

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Факультет інформатики та обчислювальної техніки

(повне найменування інституту, факультету)

Автоматизованих систем обробки інформації і управління

(повна назва кафедри)

«До захисту допущено»

В.о. завідувача кафедри

(підпис) О.А.Павлов
(ініціали, прізвище)

“ ” 2019 р.

Дипломний проект

на здобуття ступеня бакалавра

з напрямку підготовки _____ **6.050103 «Програмна інженерія»**

на тему _____ **Web – застосування для узгодження тематики дипломних проектів**

ІП-51

Виконав: студент IV курсу, групи _____ **Гохгалтер Руслан Сергійович**
(прізвище, ім'я, по батькові)

(підпис)

Керівник _____ **Доц., к.т.н., доц. Фіногенов О.Д.**

посада, науковий ступінь, вчене звання, прізвище, ініціали

(підпис)

**Консультант
з графічної
документації**

_____ **ст. викл. Головченко М.М.**

посада, науковий ступінь, вчене звання, прізвище, ініціали

(підпис)

Рецензент:

_____ **Доц., к.т.н., доц. Мелкумян К.Ю.**

посада, науковий ступінь, вчене звання, прізвище, ініціали

(підпис)

Засвідчую, що у цьому дипломному проекті
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____
(підпис)

Київ – 2019 року

**Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”**

Факультет (інститут) Інформатики та обчислювальної техніки
(повна назва)

Кафедра автоматизованих систем обробки інформації і управління
(повна назва)

Рівень вищої освіти – перший (бакалаврський)

Напрямок підготовки (програма професійного спрямування) – **6.050103**
«Програмна інженерія» (Програмне забезпечення систем)

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри

О.А. Павлов
(підпис) (ініціали, прізвище)

“ ” 2019 р.

**ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ СТУДЕНТУ**

Гохгалтеру Руслану Сергійовичу
(прізвище, ім'я, по батькові)

**1. Тема проекту « Web – застосування для узгодження тематики
дипломних проектів »**

керівник проекту Олексій Дмитрович Фіногенов, к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “23” квітня 2019 р. №1181-с

2. Термін подання студентом проекту «03» червня 2019 року

3. Вихідні дані до проекту

Технічне завдання

4. Зміст пояснювальної записки

*1) Аналіз вимог до програмного забезпечення: основні визначення та терміни,
опис предметного середовища, огляд існуючих технічних рішень та відомих*

програмних продуктів, розробка функціональних та нефункціональних вимог

*2) Моделювання та конструювання програмного забезпечення: моделювання та
аналіз програмного забезпечення, засоби розробки, технічні рішення, архітектура*

3) Аналіз якості та тестування програмного забезпечення

*4) Впровадження та супровід програмного забезпечення. Керівництво
користувача. Керівництво адміністратора.*

5. Перелік графічного матеріалу

1) *Схема структурна варіантів використань*

3) *Схема бази даних*

4) *Схема структурна класів програмного забезпечення*

6. Консультанти розділів проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання «12» березня 2018 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів проекту	Примітка
1.	<i>Вивчення рекомендованої літератури</i>		
2.	<i>Аналіз існуючих методів розв'язання задачі</i>		
3.	<i>Постановка та формалізація задачі</i>		
4.	<i>Аналіз вимог до програмного забезпечення</i>		
5.	<i>Алгоритмізація задачі</i>		
6.	<i>Моделювання програмного забезпечення</i>		
7.	<i>Обґрунтування використовуваних технічних засобів</i>		
8.	<i>Розробка архітектури програмного забезпечення</i>		
9.	<i>Розробка програмного забезпечення</i>		
10.	<i>Налагодження програми</i>		
11.	<i>Виконання графічних документів</i>		
12.	<i>Оформлення пояснювальної записки</i>		
13.	<i>Подання ДП на попередній захист</i>		
14.	<i>Подання ДП рецензенту</i>		
15.	<i>Подання ДП на основний захист</i>		

Студент _____ Гохгалтер Р.С.
(підпис)

Керівник проекту _____ Фіногенов О.Д.
(підпис)

[illegible]

АНОТАЦІЯ

Пояснювальна записка дипломного проекту складається з чотирьох розділів, містить 35 таблиць, 8 рисунків, 3 додатки та 9 джерел – загалом 61 сторінок.

Об'єкт дослідження: веб системи моделі клієнт-сервер з реляційними базами даних, що використовуються для роботи сайту для узгодження тематики дипломних проектів.

Мета дипломного проекту: створити веб сервіс, що може бути використаний для узгодження тематики дипломних проектів.

У першому розділі було проаналізовано предметну область та розроблено архітектуру для створення проекту. Побудовані таблиці використання і проаналізовані конкуренти.

У другому розділі були створені сценарії використання системи користувачем та описані класи системи.

У третьому розділі були описані критерії якості програмного забезпечення та тестування системи.

У четвертому розділі описано розгортання та впровадження системи.

У додатках наведено: інструкцію користувача, інструкцію адміністратора, діаграма класів, опис бази даних.

КЛЮЧОВІ СЛОВА: АРХІТЕКТУРА МОДЕЛЬ-ВИД-КОНТРОЛЛЕР, ВЕБ-ДОДАТОК, СЕРВІС

ABSTRAC

Explanatory note of the diploma project consists of 4 sections, 35 tables, 8 pictures, 3 annexes and 9 sources – total 61 pages.

The object of study: web-based model client-server with relational databases used to work the site to approve the topics of diploma projects.

The aim of the diploma project: create a web service that can be used to approve the topics of diploma projects.

In the first section analyzed the subject area and developed the architecture for the project. Built-in usage tables and analyzed competitors..

In the second section, user scripts were created and the system classes described

The third section describes the quality criteria of software and system testing.

The fourth section describes the deployment and implementation of the system.

The annexes contain: user manual, administrator's manual, class diagram, database description.

KEYWORDS: MODEL-VIEW-CONTROLLER ARCHITECTURE, WEB-APPLICATION, SERVICE

					КПІ.ІП-5105.045440.03.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		5

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	, СИМВОЛІВ, ОДИНИЦЬ,	
СКОРОЧЕНЬ І ТЕРМІНІВ		9
ВСТУП.....		10
1 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ		12
1.1 ЗАГАЛЬНІ ПОЛОЖЕННЯ		12
1.2 ЗМІСТОВНИЙ ОПИС І АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ		12
1.3 АНАЛІЗ УСПІШНИХ ІТ-ПРОЕКТІВ.....		13
1.4 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ		14
1.4.1 Постановка задачі		25
1.5 Висновки по розділу		25
2 МОДЕЛЮВАННЯ ТА КОНСТРУЮВАННЯ ПРОГРАМНОГО		
ЗАБЕЗПЕЧЕННЯ.....		27
2.1 МОДЕЛЮВАННЯ ТА АНАЛІЗ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....		27
2.2 АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ		33
2.3 КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ		34
2.4 ОПИС СТРУКТУРИ БАЗИ ДАНИХ		47
2.5 АНАЛІЗ БЕЗПЕКИ ДАНИХ		55
2.6 Висновки по розділу		55
3 АНАЛІЗ ЯКОСТІ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ		57
4 ВПРОВАДЖЕННЯ ТА СУПРОВІД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ		58
4.1 РОЗГОРТАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....		58
4.1.1 Встановлення Apache.....		58
4.1.2 Встановити Php 7.2.....		58
4.1.3 Встановити субд Mysql.....		58
4.1.4 Створити базу даних.....		58
4.1.5 Задати необхідні налаштування підключення до бази даних.....		58
4.2 ІНСТРУКЦІЯ КОРИСТУВАЧА		59
4.3 ІНСТРУКЦІЯ АДМІНІСТРАТОРА		59
ВИСНОВКИ		60
ПЕРЕЛІК ПОСИЛАНЬ.....		61

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

Фреймворк – бібліотека до програмної мови, що полегшую розробку програмних продуктів.

Система – програмний застосунок, що розроблюється для виконання теми дипломного проекту.

Токен – набір випадкових чисел, що використовується для ідентифікації запитів, що надходять до серверу.

					КПІ.ІП-5105.045440.01.81	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

ВСТУП

Для навчання нових поколінь в університетах, найважливішим, на мою думку, є комунікація з викладачами. В університетах викладачі не тільки навчають студентів, а й займаються різноманітною науковою діяльністю. Над деякими проектами викладачі працюють десятиліттями і важливим є залучення до цих проектів студентів. Це не тільки прискорює викладачів, але й допомагає у навчанні студентам. Багато професорів та аспірантів залучають студентів у свої роботи через написання дипломів та курсових робіт. У деяких випадках співпраця продовжується і після університету, коли проекти отримують гранти і інвесторів.

Але з розвитком інформаційних технологій, у студентів змінились способи комунікації і сприйняття інформації. Нове покоління звикло до пошуку інформації у інтернеті і більше не дивиться на дошки оголошень в університеті і не звертається з питаннями до деканату. Через це, багато викладачів не можуть роками знайти помічників, а студентам важко визначити тему диплому, або навіть курсової роботи. Саму тому потрібно створити веб-застосунок, де викладачі можуть розмістити теми дипломів та курсових робіт, а студенту підібрати задачу по силам.

Також ця система може знадобитись не тільки викладачам, а й іншим працівникам КПІ, які зіткнулись із монотонною роботою, яку можна автоматизувати.

Ця система повинна мати мінімалістичний дизайн та інтерфейс, що буде зрозумілий кожному. Коли студент знаходить цікаву задачу, він може надіслати запит на виконання цієї задачі викладачу, а той в свою чергу, прийняти запит, надіславши свою контакти студенту.

Цілі:

- надати викладачам, студентам зручний сервіс, який дозволить розмістити та затвердити теми дипломного проекту;
- допомогти методистам повідомити про не автоматизовані задачі.

					КПІ.ІП-5105.045440.01.81	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

Задачі:

- створення, редагування, видалення теми диплому викладачем;
- створення, редагування, видалення задачі методистом;
- створення теми диплома викладачем, на основі задачі методиста;
- відправка запиту на затвердження теми диплома;
- утвердження теми диплома.

					КПІ.ІП-5105.045440.01.81	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

1 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1.1 Загальні положення

Через постійний аналіз дій користувачів, подібний програмний продукт постійно модифікується та змінюється. Потрібно обрати фреймворк, що має реалізацію загальних алгоритмів та який дозволить швидко розширювати можливості продукту. Також потрібно розробити архітектуру проекту, що зможе легко та швидко розширюватись.

1.2 Змістовний опис і аналіз предметної області

Платформа для розміщення завдань та знаходження виконувачів цих завдань – це сучасна дошка оголошень де користувачі діляться на різні ролі та мають різний функціонал і цілі. Постановники задач описують задачу, необхідні навички для виконання та дедлайн для виконання. Далі він розміщує це завдання у загальному списку задач і чекає, поки виконавець не відправить запит на виконання. В деяких випадках постановник може сам шукати люди, що виконає завдання. Після знаходження виконавця, постановник більш детально описує умови та відповідає на різні питання, далі він затверджує за виконавцем задачу. Виконувач може створювати запити на виконання та передивлятися список задач.

Список задач представлений у вигляді сторінки з пагінацією, да в самому початку розміщена найновіша задача, а в кінці останньої сторінки найстаріша. Для кожної задачі вказується тема, опис, дата створення, необхідні навички для виконання.

Також кожен користувач повинен мати діалогове вікно, де він зможе передивитись статус запитів, на виконання задачі.

1.3 Аналіз успішних ІТ-проектів

Серед подібних систем можна виділити сайти для фрілансерів та майстрів на всі руки. В цих сервісах зареєстровані виконавчі, що на постійній основі шукають та виконують завдання за грошову винагороду. Зазвичай серед фрілансерів створюється конкуренція, що дозволяє постанувальнику обрати оптимального виконавця за ціною роботи і вміннями.

Kabanchik – цей сайт призначений для розміщення завдань будь-якого типу та пошуку виконавців цих завдань. Як і веб-додаток для узгодження тематики дипломних проектів, цей сайт розділяє користувачів сайту на виконавців і роботодавців, але відмінністю є не тільки тематика завдань, а й алгоритм прийняття завдань. На “кабанчику” набагато більше виконавців ніж роботодавців і в деяких ситуаціях на виконання завдання може претендувати більше 100 чоловік.

freelancehunt – сайт призначений для пошуку виконавця. На цьому сайті акцент йде представленні списку виконавців, щоб роботодавці могли знайти виконавця, не створюючи завдання. Виконавці представленні у вигляді рейтингового списку, чим більше рейтингу у виконавця, тим більше у нього шансів знайти для себе роботу. Але проект з пошуку та затвердженню тем дипломів акцентує увагу саме на розміщенні завдання.

Weblancer – сервіс, що призначений для пошуку програмістів для вирішення питань пов'язаних з інформаційними технологіями у мережі інтернет. На відміну від багатьох схожих сайтів, цей сайт акцентує увагу саме на конкретній предметній області.

Але серед подібних сайтів та платформ не має сервісів для пошуку та затвердження дипломних проектів. Деякі задачі, що розміщуються на цих сервісах описані непрофесійно і незрозуміло, дуже часто потребується використовувати застарілі технології для їх вирішення. Але викладачі, що мають великий академічний опит, завжди описують задачу чітко та зрозуміло. Тому задачі, що будуть створенні методистами КПІ, повинні спершу пройти

					КПІ.ІП-5105.045440.01.81	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

модерацію викладача, перед тим, як потрапити до списку тем дипломних проектів. Студенту залишається лише створити запит на виконання та чекати рішення викладача.

1.4 Аналіз вимог до програмного забезпечення

Система взаємодіє з такими учасниками: студент, викладач, методист, адміністратор.

Варіанти використання наведені в таблицях:

Таблиця 1.1 – Варіант використання UC001

Назва	Реєстрація
Опис	Користувач реєструється у системі
Учасники	Студент, викладач, методист, адміністратор
Передумови	Користувач не зареєстрований
Постумови	Успішна реєстрація
Основний сценарій	Система створює форму реєстрації, після відправлення форми реєструє користувача
Розширення сценаріїв	4.1 Дані не проходять валідацію 4.1.a Система вказує користувачу на неправильно заповнені поля реєстрації

Таблиця 1.2 – Варіант використання UC002

Назва	Авторизація
Опис	Авторизація зареєстрованого користувача на сайті
Учасники	Студент, викладач, методист, адміністратор

Продовження таблиці 1.2

Передумови	Користувач зареєстрований
Постумови	Створення сесії користувача
Основний сценарій	Система створює форму авторизації, після відправки форми авторизує користувача
Розширення сценаріїв	4.1 Дані не проходять валідацію 4.1.а Система вказує користувачу на неправильно заповнені поля форми авторизації

Таблиця 1.3 – Варіант використання UC003

Назва	Вихід з облікового запису
Опис	Користувач виходить з облікового запису
Учасники	Викладач, студент, методист, адміністратор
Передумови	Користувач авторизований в системі
Постумови	Користувач вийшов з системи
Основний сценарій	Користувач натискає на кнопку «Вийти»
Розширення сценаріїв	

Таблиця 1.4 – Варіант використання UC004

Назва	Створення власної теми дипломного проекту
Опис	Викладач створює власну тему дипломного проекту
Учасники	Викладач

Продовження таблиці 1.4

Передумови	Викладач авторизований
Постумови	У список тем дипломів додається нова тема
Основний сценарій	Викладач натиснув на «Створити тему дипломного проекту». Система створює форму для створення теми дипломного проекту, викладач заповнює форму й натискає на кнопку «Відправити». Далі система зберігає тему диплому.
Розширення сценаріїв	4.1 Дані не проходять валідацію 4.1.а Система вказує викладачу на неправильно заповнені поля форми

Таблиця 1.5 – Варіант використання UC005

Назва	Створення дипломного проекту на основі завдання методиста
Опис	Викладач створює тему дипломного проекту на основі завдання методиста
Учасники	Викладач, методист
Передумови	Методист створив завдання, викладач авторизований
Постумови	У список тем дипломів додається нова тема

Продовження таблиці 1.5

Основний сценарій	Викладач зайшов на сторінку завдання методиста і натиснув на кнопку «Створити тему дипломного проекту». Система на основі задачі методиста створює форму для створення теми дипломного проекту з заповненим полем «Тема диплому» та заповненим полем «Опис». Вказані заповнені поля містять дані з задачі методиста
Розширення сценаріїв	4.1 Дані не проходять валідацію 4.1.а Система вказує викладачу на неправильно заповнені поля форми

Таблиця 1.6 – Варіант використання UC006

Назва	Створення задачі
Опис	Методист створює задачу
Учасники	Методист
Передумови	Методист авторизований
Постумови	Задача створена
Основний сценарій	Методист натискає на кнопку «Створити задачу». Система створює форму для створення задачі. Методист заповнює поля та натискає на кнопку «Створити». Система зберігає задачу та додає її у список задач від методистів.

Продовження таблиці 1.6

Розширення сценаріїв	4.1 Дані не проходять валідацію 4.1.а Система вказує викладачу на неправильно заповнені поля форми
----------------------	---

Таблиця 1.7 – Варіант використання UC007

Назва	Створення запиту на затвердження теми диплому
Опис	Студент обирає тему диплому та створює запит на затвердження теми диплому
Учасники	Студент, викладач
Передумови	Викладач створив тему диплому, студент авторизований
Постумови	Студент відправив запит
Основний сценарій	Студент вибирає тему диплома зі списку тем дипломів та натискає на кнопку «Відправити запит на затвердження теми диплому»
Розширення сценаріїв	

Таблиця 1.8 – Варіант використання UC008

Назва	Перегляд статусів запитів на затвердження теми диплома
-------	--

Продовження таблиці 1.8

Опис	Користувач може переглядати статус запитів затвердження теми диплому
Учасники	Викладач, студент, методист
Передумови	Створена тема диплому
Постумови	Користувач дізнався про статус запитів
Основний сценарій	Користувач натискає на кнопку «Запити». Система формує список запитів на затвердження тем дипломів та їхні статуси, далі система виводить список користувачу
Розширення сценаріїв	

Таблиця 1.9 – Варіант використання UC009

Назва	Затвердження теми диплома
Опис	Викладач затверджує тему диплома
Учасники	Викладач
Передумови	Викладач зареєстрований. Студент створив запит на затвердження теми диплома
Постумови	Оновився статус запиту на затвердження теми диплому

Продовження таблиці 1.9

Основний сценарій	Викладач обирає запит на затвердження теми диплому та натискає на кнопку «Затвердити». Далі система оновлює статус запиту.
Розширення сценаріїв	4.1 Викладач натиснув на кнопку «Відхилити» 4.1.а Система оновлює статус запиту

Таблиця 1.10 – Варіант використання UC010

Назва	Підтвердження статусу користувача
Опис	Адміністратор підтверджує статус користувача
Учасники	Адміністратор, викладач, методист
Передумови	Методист, або викладач зареєструвались
Постумови	Адміністратор підтвердив або відхилив статус методиста, викладача
Основний сценарій	Адміністратор натиснув на кнопку «Управління користувачами» та обрав користувача. Далі адміністратор, або підтверджує статус користувача, або відхиляє
Розширення сценаріїв	

Таблиця 1.11 – Варіант використання UC011

Назва	Видалення користувача
-------	-----------------------

Продовження таблиці 1.11

Опис	Адміністратор може видаляти користувачів
Учасники	Адміністратор, викладач, методист, студент
Передумови	Викладач, студент або методист зареєстрований в системі. Адміністратор авторизований
Постумови	Викладач, адміністратор або методист видалений з бази даних
Основний сценарій	Адміністратор заходить до сторінки управління користувачами та обирає користувача. Далі адміністратор натискає н кнопку «Видалити». Система видаляє користувача з бази даних
Розширення сценаріїв	

Таблиця 1.12 – Варіант використання UC012

Назва	Блокування користувача
Опис	Адміністратор може блокувати користувачів
Учасники	Адміністратор, викладач, методист, студент
Передумови	Викладач, студент або методист зареєстрований в системі. Адміністратор авторизований
Постумови	Викладач, адміністратор або методист заблокований в системі та не може авторизуватись

Продовження таблиці 1.12

Основний сценарій	Адміністратор заходить до сторінки управління користувачами та обирає користувача. Далі адміністратор натискає на кнопку «Блокувати». Система блокує користувача
Розширення сценаріїв	

Таблиця 1.13 – Варіант використання UC013

Назва	Розблокування користувача
Опис	Адміністратор може розблокувати користувачів
Учасники	Адміністратор, викладач, методист, студент
Передумови	Викладач, студент або методист заблокований в системі. Адміністратор авторизований
Постумови	Викладач, адміністратор або методист розблокований в системі
Основний сценарій	Адміністратор заходить до сторінки управління користувачами та обирає заблокованого користувача. Далі адміністратор натискає на кнопку «Розблокувати». Система розблоковує користувача
Розширення сценаріїв	

Функціональні вимоги:

Таблиця 1.14 – Опис функціональної вимоги REQ001

Номер	REQ001
Назва	Створення тем дипломних проектів
Опис	Система дозволяє створювати теми дипломних проектів викладачам.

Таблиця 1.15 – Опис функціональної вимоги REQ002

Номер	REQ002
Назва	Затвердження теми дипломного проекту
Опис	Система дає можливість студенту створити запит на затвердження теми дипломного проекту, а викладачу в свою чергу затвердити тему

Таблиця 1.16 – Опис функціональної вимоги REQ003

Номер	REQ003
Назва	Управління користувачами
Опис	Система дозволяє адміністратору керувати користувачами, а саме видалити, блокувати, розблоковувати, підтверджувати статус користувачів

Таблиця 1.17 – Опис функціональної вимоги REQ004

					КПІ.ІП-5105.045440.01.81	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		23

Номер	REQ004
Назва	Створення задачі

Продовження таблиці 1.17

Опис	Система надає можливість методисту створити задачу ментору.
------	---

Таблиця 1.18 – Опис функціональної вимоги REQ005

Номер	REQ005
Назва	Реєстрація користувачів
Опис	Користувач може реєструватись в системі

Таблиця 1.19 – Опис функціональної вимоги REQ006

Номер	REQ006
Назва	Авторизація користувачів
Опис	Користувач може авторизуватись в системі

Таблиця 1.20 – Опис функціональної вимоги REQ007

Номер	REQ007
Назва	Вихід з облікового запису
Опис	Користувач може вийти з облікового запису

Результуюча матриця трасування зображення на рисунку - рисунок 1.1.

	UC001	UC002	UC003	UC004	UC005	UC006	UC007	UC008	UC009	UC010	UC011	UC012	UC013
REQ001 Створення тем дипломних проектів													
REQ002 Затвердження теми дипломного проекту													
REQ003 Управління користувачами													
REQ004 Створення задач													
REQ005 Реєстрація користувачів													
REQ006 Авторизація користувачів													
REQ007 Вихід з облікового запису													

Рисунок 1.1 - матриця трасування

1.4.1 Постановка задачі

Розробка призначена для студентів, які не можуть знайти теми дипломного проекту, викладачам, якій мають теми дипломних проектів та методистам, яким потрібно автоматизувати процеси.

Метою розробки є спрощення утвердження та пошуку дипломних проектів студентам, створення тем дипломних проектів на основі задач, які повинні допомогти автоматизувати роботу з інформацією в університеті, шляхом створення web – додатку, який має функціонал для розміщення та затвердження тем дипломних проектів.

1.5 Висновки по розділу

У розділі була проаналізована предметна область програмного продукту та розглянуті успішні іт – проекти, пов’язані з предметною областю. Були описані функціональні вимоги та побудована матриця трасування.

					КПІ.ІП-5105.045440.01.81	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

2 МОДЕЛЮВАННЯ ТА КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Моделювання та аналіз програмного забезпечення

Для розуміння роботи алгоритмів бізнес процесів та їх успішної інтеграції у програмний код, потрібно побудувати діаграми у форматі BPMN, де будуть описані сценарії використання системи користувачем.

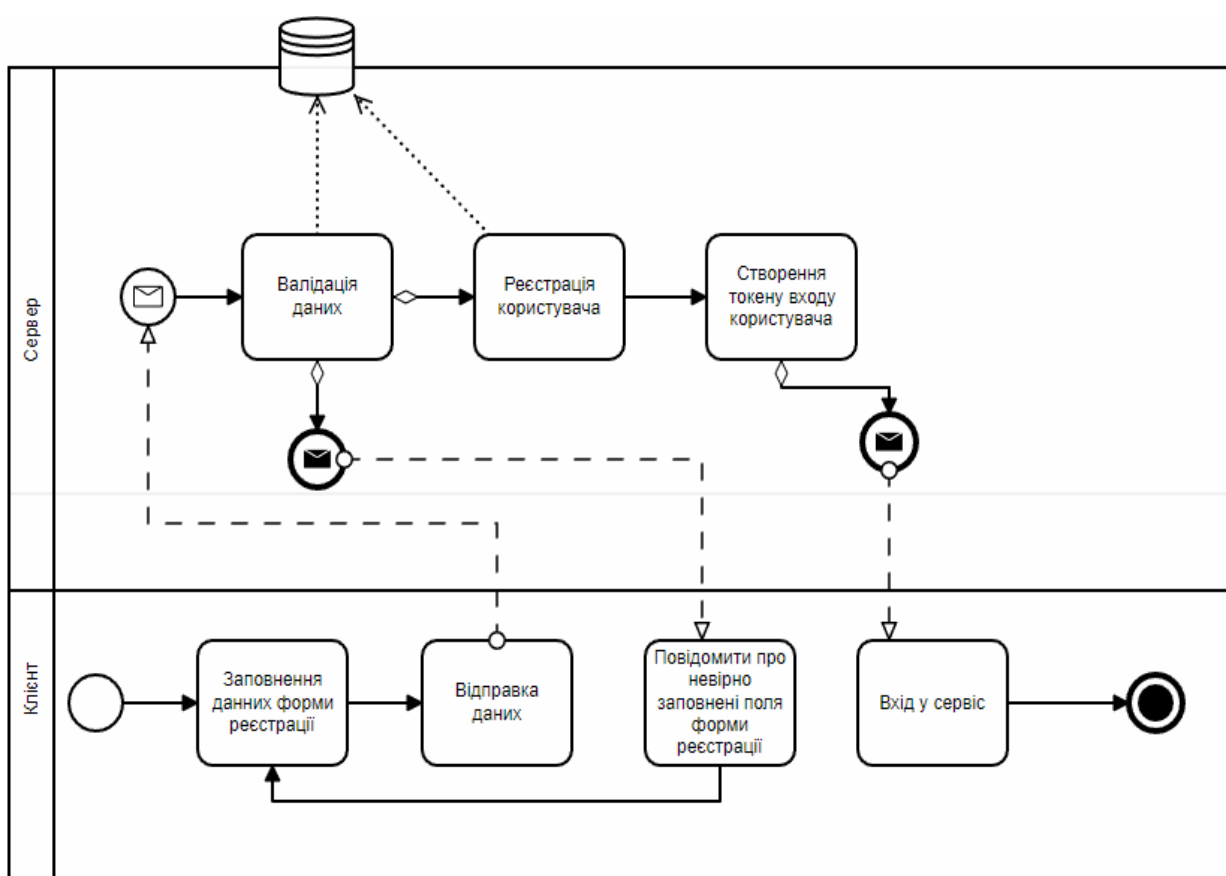


Рисунок 2.1 – Реєстрація користувача

Опис процесу реєстрації користувача:

- користувач заповнює форму реєстрації;
- користувач відправляє дані;
- дані відправляються на сервер;
- далі розроблений алгоритм валідує дані;
- якщо валідація даних не успішна, система повідомляє користувача;

- інакше система реєструє користувача;
- далі система створює токен входу для користувача;
- далі система відсилає користувачу токен.
- користувач отримує токен та входить у сервіс;

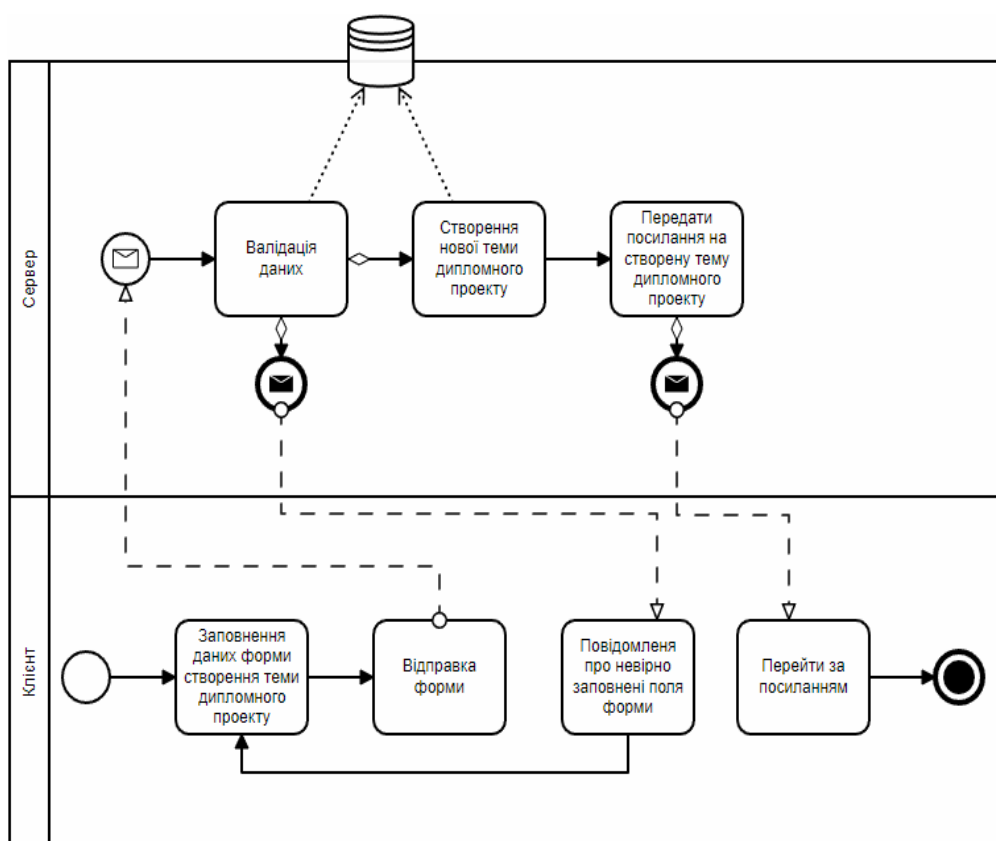


Рисунок 2.2 – Створення теми диплому

Опис процесу створення теми диплому:

- користувач заповнює поля форми створення теми диплому;
- користувач відправляє дані;
- дані відправляються на сервер;
- якщо валідація даних не успішна, система повідомляє користувача;
- інакше система створює нову тему диплома;
- далі система передає посилання на нову тему диплому;
- користувач перенаправляється на сторінку створеної теми диплому.

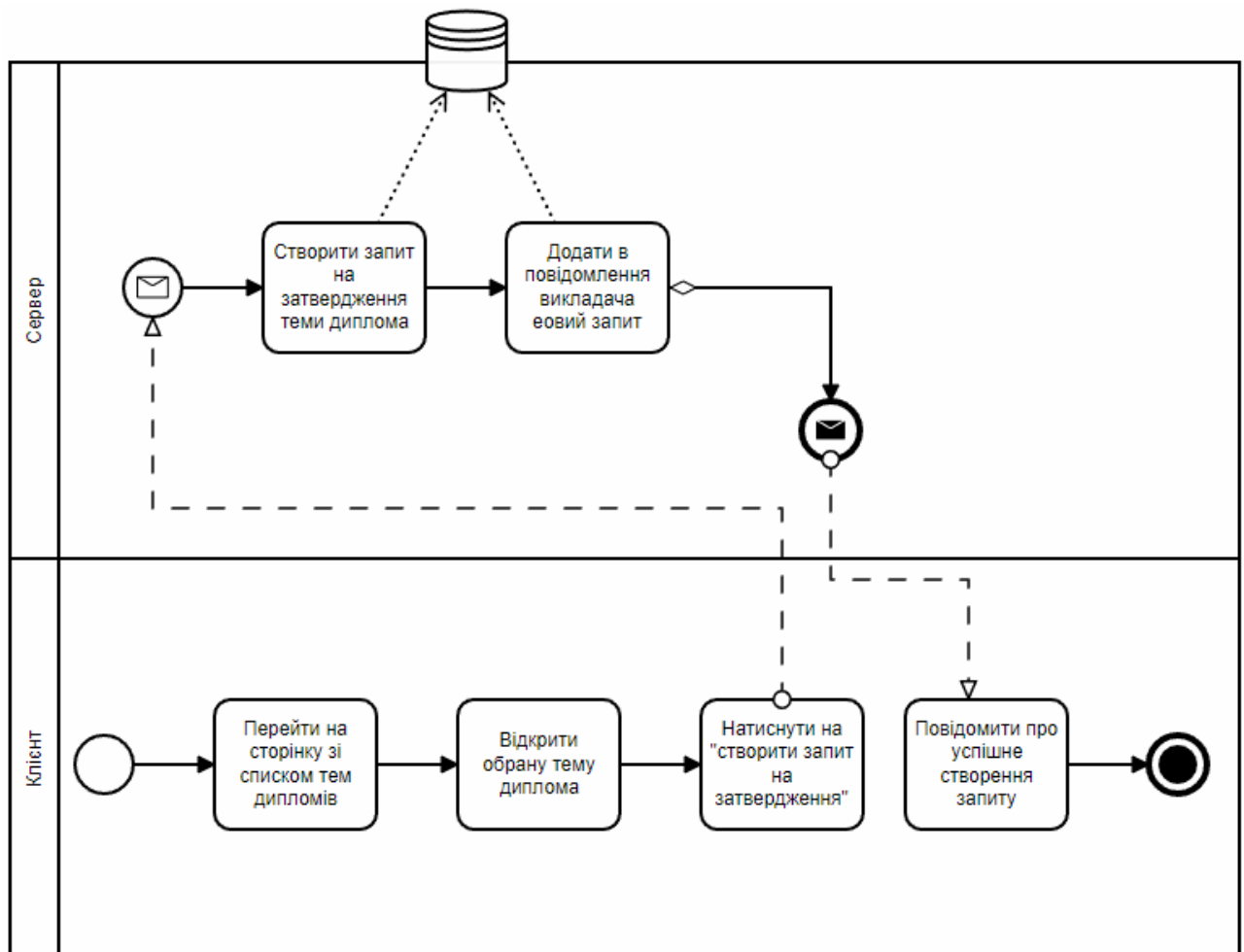


Рисунок 2.3 – Запит на затвердження теми диплому

Опис процесу створення запиту на затвердження теми диплому:

- користувач переходить на сторінку зі списком тем дипломів;
- користувач вибирає та відкриває тему диплому;
- користувач натискає на «Створити запит на затвердження»;
- запт відправляється на сервер;
- система створює запит на затвердження;
- система створює повідомлення для викладача про запит;
- система повідомляє користувача про успішно надісланий запит.

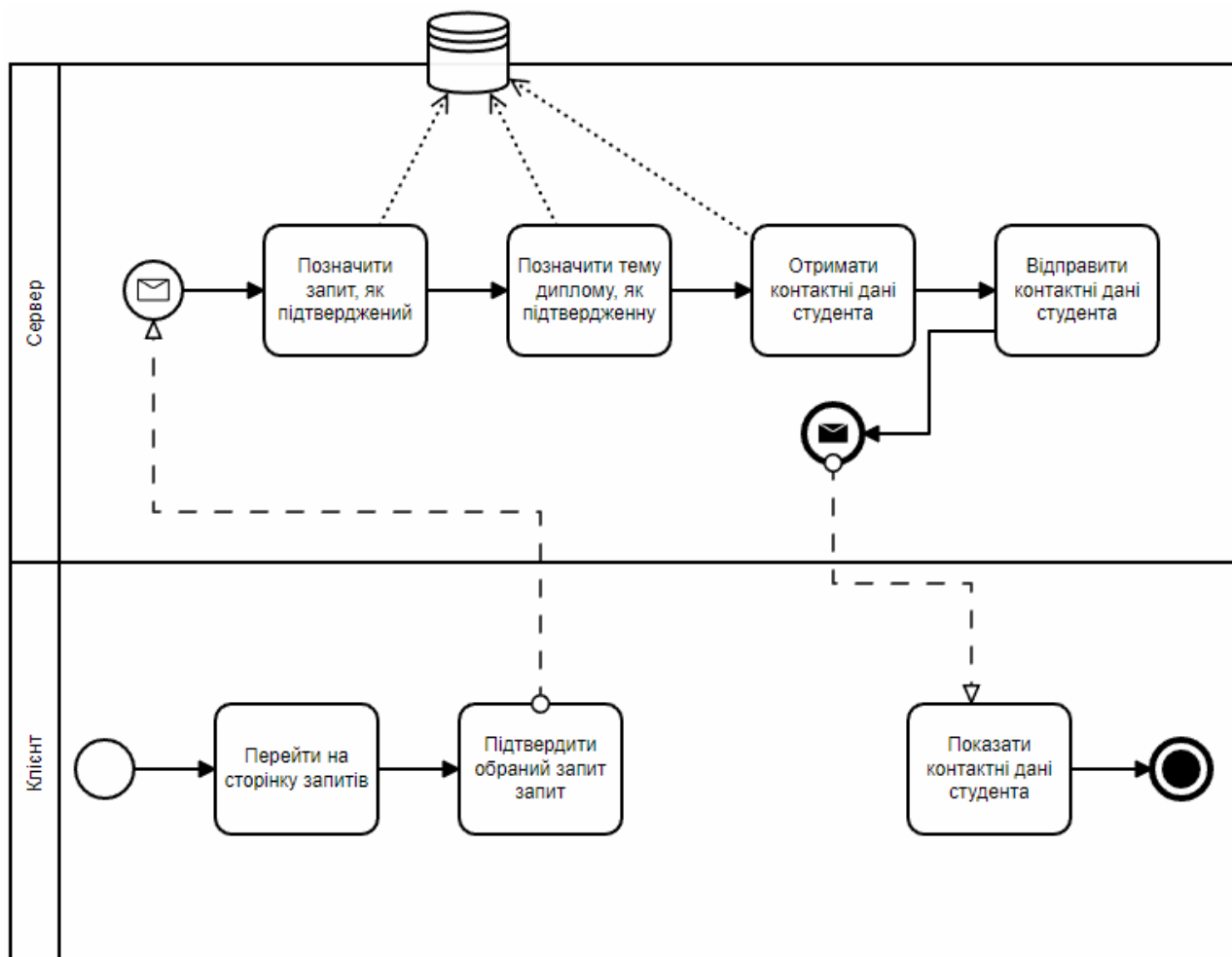


Рисунок 2.4 – Затвердження теми диплома

Опис затвердження теми диплома:

- користувач переходить на сторінку запитів;
- користувач підтверджує запит на затвердження теми диплома;
- дані відправляються до серверу;
- система позначає запит, як підтверджений;
- система позначає тему диплому, як підтверджену;
- система знаходить контактні дані студента, що відправив запит на затвердження;
- система відправляє дані студента користувачу;
- користувач отримує контактні дані студента.

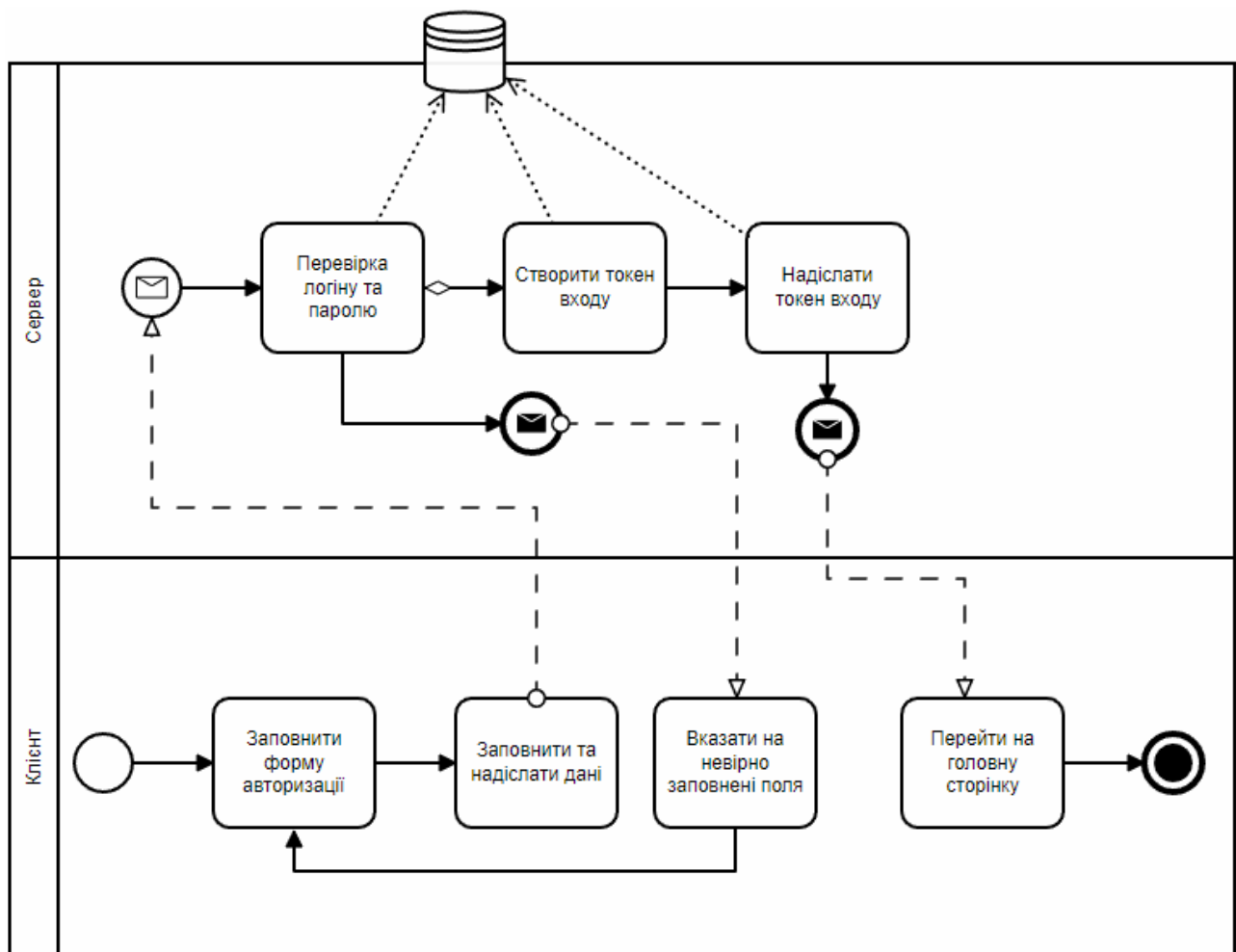


Рисунок 2.5 – Авторизація користувача

Опис авторизації користувача:

- користувач заповнює форму авторизації;
- користувач надсилає дані;
- дані відправляються на сервер;
- система перевіряє логін і пароль. Якщо валідація не успішна, система надсилає повідомлення користувачу;
- інакше система створює токен входу;
- далі система надсилає токен до користувача;
- користувач отримує токен та переходить на головну сторінку.

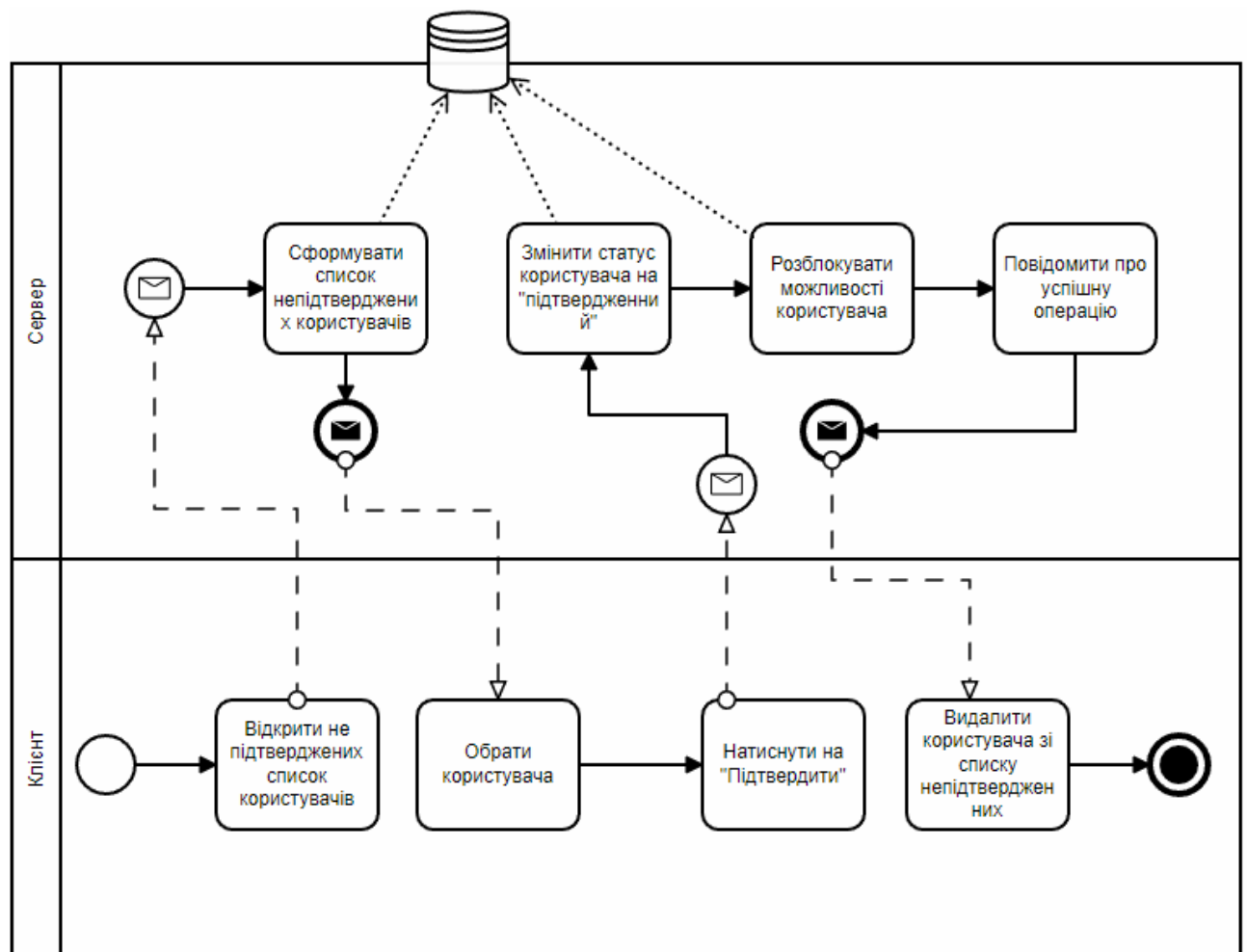


Рисунок 2.6 – Підтвердження статусу користувача

Наведено послідовний опис процесу реєстрації користувача:

- адміністратор відкриває список не підтверджених користувачів;
- сервер формує список непідтверджених користувачів то відсилає до адміністратора;
- адміністратор обирає користувача;
- адміністратор натискає на кнопку «Підтвердити»;
- запит відправляється на сервер;
- система отримує запит та змінює статус користувача на підтверджений;
- система розблоковує користувача;

- система надсилає повідомлення про успішну зміну статусу користувача.
- адміністратор отримує повідомлення, користувач видаляється зі списку непідтверджених;

2.2 Архітектура програмного забезпечення

Система створена з використанням архітектури MVC, яка дозволяє відокремити роботу з даними і представленнями. Архітектура MVC дозволяє швидко змінювати та розширювати систему, що необхідно у розробці подібних сайтів.

В якості веб серверу був обраний Apache. Цей веб-сервер має відкритий код та розробляється з 1995 року і зарекомендував себе, як надійний і швидкий веб-сервер.

Для клієнтської частини буде використовуватись мова гіпертекстової розмітки – HTML, мова для виконання клієнтських скриптів Javascript та CSS – мова для описання стилей. Також для кращої розробки скриптів для клієнтської частини буде використовуватись jQuery. JQuery – це бібліотека для мови Javascript, що дозволяє писати набагато зрозуміліший та коротший код.

На серверній частині буде використовуватись мова php 7.2. Це найпопулярніша мова в світі, що використовуються для написання серверної частини найрізноманітніших веб-застосунків. Також php є однією з найшвидкіших мов для роботи веб-застосунків і має вбудований збирач сміття. Також php добре оптимізований для роботи з великими масивами і має багато вбудованих команд, що прискорить розробку.

В якості фреймворку для серверної частини був обраний yii2. Yii2 – це фреймворк для php, який переважно розробляють в Україні і це один з небагатьох фреймворків, який має документацію на українській мові. Також

використання yii2 обумовлене необхідністю розширювати систему в майбутньому, а цей фреймворк має багато готових рішень для розширення.

2.3 Конструювання програмного забезпечення

Модель MVC передбачає розділення логіки моделей і контролерів, тож класи переважно розділені на контролери та моделі.

Таблиця 2.1 – Опис класів – контролерів системи

Клас	Опис
AdminController	Клас, що описує функції адміністратора
DepartamentController	Клас, що описує функції роботи зі списком кафедр
DialogController	Клас, що логіку роботи запитів на затвердження тем диплому
DiplomaController	Клас, що описує логіку роботи створення та управління темами дипломних робіт
LoginController	Клас, що описує логіку роботи входу користувачів до особистого кабінету.
SignUpController	Клас, що описує логіку роботи форми реєстрації
SiteController	Клас, що описує основні функції системи

Таблиця 2.2 – Опис класів – моделей системи

Клас	Опис
Departament	Модель, що описує таблицю кафедр

Продовження таблиці 2.2

Клас	Опис
Dialog	Модель, що описую таблицю діалго та формує списки статусів запитів на затвердження дипломів
Diploma	Модель, що описує таблицю дипломів, та управляє статусами цих дипломів
DiplomaForm	Клас, що описує форму створення диплому, правила валідації полів та логіку збереження
DiploamSearch	Клас, що описує логіку формування списку дипломів
Faculty	Клас, що описує таблицю факультетів і правила валідація створення цих факультетів
Group	Клас, що описує таблицю груп і містить правила валідації при створенні нової групи
LoginFrom	Клас, що описує форму авторизації та правила валідації полів цієї форми
ParticleDiplomaForm	Клас, що описує форму створення задачі для методистів та правила валідації полів цієї форми
RequestForm	Клас, що описує форму створення запитів для затвердження тем дипломних проектів

Продовження таблиці 2.2

Клас	Опис
SignUpForm	Абстрактний клас, що описує базові поля та валідацію для форми реєстрації
Student	Клас, що Описує таблицю студентів та методи роботи з даними
StudentSignUpForm	Клас, що описує логіку роботи форми реєстрації студента та валідацію полів
Teacher	Клас, що описує таблицю викладачів та методи роботи з даними
TeacherSignUpForm	Клас, що описує логіку роботи форми реєстрації викладача та валідацію полів
UpdateDiplomaForm	Клас, що описує логіку роботи створення теми дипломного проекту на основі задачі від методиста
User	Клас, що описує логіку роботи користувача, його ідентифікацію та роботу з даними
UserSearch	Клас, що описує логіку формування списку користувачів, зокрема користувачів, статус яких не підтверджений

Продовження таблиці 2.2

Клас	Опис
Worker	Клас, що описує таблицю методистів та методи роботи з даними
WorkerSignUpForm	Клас, що описує логіку роботи форми реєстрації методиста та валідацію полів

Розглянемо детальніше кожну функцію по всім класам та функціям в таблиці, що буде наведена нижче.

Таблиця 2.3 – Опис методів класів та інтерфейсів системи

Клас/Модуль	Метод	Опис
AdminController	behaviors ()	Метод описує правила доступу до методів класу
AdminController	actionIndex ()	Методи описує створення та вивід списку користувачів
DepartmentController	actionTakeDepartaments()	Метод описує формування списку кафедр
DialogController	Behaviors()	Правила доступу до методів класу

Продовження таблиці 2.3

Клас/Модуль	Метод	Опис
DialogController	actionIndex()	Вивід списку запитів для користувача
DialogController	actionCreateRequest()	Створення нового запиту
DialogController	actionChangeRequestStatus()	Зміна статусу запиту
DiplomaController	Behaviors()	Описання правила доступу до методів класу
DiplomaController	actionIndex (id)	Відкриття теми диплома
DiplomaController	actionCreate ()	Створення теми диплома
DiplomaController	actionUpdate ()	Оновлення теми диплома, або створення теми диплома на основі задачі методиста
DiplomaController	actionMyList ()	Вивід списку тем дипломних робіт конкретного викладача
DiplomaController	actionCreateParticle ()	Створення задачі методиста
DiplomaController	actionParticle (\$id)	Відкриття задачі методиста

Продовження таблиці 2.3

Клас/Модуль	Метод	Опис
DiplomaController	actionSearch ()	Вивід списку тем дипломів
DiplomaController	actionSearchParticle ()	Вивід списку задач методиста
LoginController	behaviors()	Описання правила доступу до методів класу
LoginController	actionIndex()	Авторизація до системи
LoginController	actionLogout()	Вихід з облікового запису
SignUpController	behaviors()	Описання правила доступу до методів класу
SignUpController	actionIndex()	Відкриття сторінки вибору форми реєстрації
SignUpController	actionStudent()	Форма реєстрації студента
SignUpController	actionTeacher()	Форма реєстрації викладача
SignUpController	actionWorker()	Форма реєстрації методиста
SiteController	behaviors()	Описання правила доступу до методів класу

Продовження таблиці 2.3

Клас/Модуль	Метод	Опис
SiteController	actionIndex()	Відкриття головної сторінки сайту
Departament	tableName()	Отримання назви таблиці кафедр
Dialog	tableName()	Отримання назви таблиці запитів
Dialog	createTeacherDialog()	Отримання списку запитів для викладача
Dialog	createStudentDialog()	Отримання списку запитів для студента
Dialog	createWorkerDialog()	Отримання списку запитів для методиста
Dialog	exist(\$userId, \$diplomaId)	Перевірка автора теми диплому
Dialog	needHide(\$diplomaId, User \$user)	Перевірка необхідності виводі кнопок дій для користувача
Dialog	changeStatus(\$id, \$status)	Зміна статусу запиту на тему диплому

Продовження таблиці 2.3

Клас/Модуль	Метод	Опис
Dialog	declineAll(\$diploma_id)	Відхилити всі запити на затвердження конкретної теми диплому
Diploma	formName()	Зміна назви форми за надібністю
Diploma	tableName()	Отримати назву таблиці тем дипломів
Diploma	createList()	Створення списку тем дипломів
Diploma	createParticleList()	Створення списку задач методистів
Diploma	changeStatus(\$id, \$status)	Зміна статусу конкретного диплому
Diploma	getDiplomasByTeacher()	Отримати список тем дипломів конкретного викладача
DiplomaForm	rules()	Правила валідації форми створення тем дипломів
DiplomaForm	saveData()	Валідація та збереження нової теми диплому

Продовження таблиці 2.3

Клас/Модуль	Метод	Опис
Faculty	tableName()	Отримання назви таблиці факультетів
Group	tableName()	Отримання назви таблиці груп
LoginForm	rules()	Правила валідації форми авторизації на сайті
LoginForm	validatePassword(\$attribute, \$params)	Валідації пароля
LoginForm	login()	Створення токена входу для користувача
LoginForm	getUser()	Знайти користувача за допомогою його пошти
ParticleDiplomaForm	rules()	Правила валідації форми створення задачі від методиста
ParticleDiplomaForm	saveData()	Валідація та збереження задачі від методиста

Продовження таблиці 2.3

Клас/Модуль	Метод	Опис
RequestForm	rules()	Правила валідації форми створення запита на затвердження теми диплому
RequestForm	saveData()	Збереження запиту
SignUpForm	rules()	Базові правила валідації для форми реєстрації
Student	tableName()	Отримати назву таблиці студентів
StudentSignUpForm	rules()	Правила валідації для форми реєстрації студента
StudentSignUpForm	signUp()	Базова функція для збереження реєстраційних даних користувача
Teacher	getTableName()	Отримати назву таблиці викладачів

Продовження таблиці 2.3

Клас/Модуль	Метод	Опис
TeacherSignUpForm	rules()	Правила валідації для форми реєстрації викладача
TeacherSignUpForm	signUp()	Функція для збереження реєстраційних даних викладача
UpdateDiplomaForm	rules()	Правила валідації для оновлення теми диплому
UpdateDiplomaForm	saveData()	Метод для оновлення теми диплому
UpdateDiplomaForm	afterSaveData(\$worker_id)	Створення теми диплому на основі задачі методиста
User	tableName()	Отримати назву таблиці користувачів
User	behaviors()	Правила доступу до методів класу
User	ifStudent()	Перевірка ролі користувача, як студента

Продовження таблиці 2.3

Клас/Модуль	Метод	Опис
User	ifTeacher()	Перевірка ролі користувача, як викладача
User	ifWorker()	Перевірка ролі користувача, як методиста
User	beforeDelete()	Видалення запитів, тем дипломів користувача перед видаленням користувача
User	getStudent()	Опис зв'язку таблиці користувачів з таблицею студентів
User	getWorker()	Опис зв'язку таблиці користувачів з таблицею методистів
User	getTeacher()	Опис зв'язку таблиці користувачів з таблицею викладачів

Продовження таблиці 2.3

Клас/Модуль	Метод	Опис
User	findIdentity(\$id)	Отримати об'єкт користувача за допомогою id
User	findIdentityByAccessToken(\$token, \$type = null)	Отримати об'єкт користувача за допомогою токена входу
User	findByEmail(\$email)	Отримати об'єкт користувача за електронної пошти
User	getId()	Отримати ід користувача
User	getAuthKey()	Отримати токен входу користувача
User	validateAuthKey(\$authKey)	Перевірити токен входу користувача
User	validatePassword(\$password)	Перевірити правильність введеного пароля користувача
User	setPassword(\$password)	Встановити пароль користувачу
User	generateAuthKey()	Згенерувати токен входу для користувача

Продовження таблиці 2.3

Клас/Модуль	Метод	Опис
UserSearch	createDataProvider()	Сформувати список користувачів
Worker	getTableName()	Отримати назву таблиці методистів
WorkerSignUpForm	rules()	Правила валідації для форми реєстрації методиста
WorkerSignUpForm	signUp()	Збереження даних з форми реєстрації методиста

2.4 Опис структури бази даних

У таблицях 2.4 – 2.13 описані таблиці бази даних.

Таблиця user призначена для зберігання даних користувача.

Таблиця 2.4 – Опис таблиці user

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
user	id	int	Код, під яким користувач зареєстрований в базі даних

Продовження таблиці 2.4

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
user	role	int	Код, що вказує на роль користувача
	status	int	Кож, що вказує на статус користувача
	created_at	int	Дата створення користувача
	updated_at	int	Дата останнього оновлення даних користувача
	name	varchar	Ім'я користувача
	surname	varchar	Прізвище користувача
	gender	int	Код, що вказує на стать користувача
	phone	varchar	Телефон користувача
	email	varchar	Електронна адреса користувача
	password_token	varchar	Токен, що зберігає код для відновлення пароля користувача
	password	varchar	Пароль користувача
	auth_key	varchar	Токен сесії користувача

Таблиця student призначена для зберігання даних студента. За допомогою ідентифікатора group_id, таблиця пов'язана з таблицею group, зв'язком один до

багатьох. За допомогою ідентифікатора `user_id`, таблиця пов'язана з таблицею `user`, зв'язком один до одного.

Таблиця 2.5 – Опис таблиці `student`

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
student – таблиця, що вказує на дані студента	id	int	Ідентифікатор студента
	user_id	int	Код, що пов'язує таблицю студентів з таблицею користувачів
	group_id	int	Код, що пов'язує таблицю студентів з таблицею груп
	curse	int	Поточний курс студента

Таблиця `teacher` призначена для зберігання даних викладача. За допомогою ідентифікатора `departament_id`, таблиця пов'язана з таблицею `departament`, зв'язком один до багатьох. За допомогою ідентифікатора `user_id`, таблиця пов'язана з таблицею `user`, зв'язком один до одного.

Таблиця 2.6 – Опис таблиці teacher

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
teacher – таблиця, що вказує на дані викладача	id	int	Ідентифікатор викладача
	user_id	int	Код, що пов'язує таблицю викладачів з таблицею користувачів
	departament_id	int	Код, що пов'язує таблицю студентів з таблицею кафедр
	degree	varchar	Вчене звання викладача

Таблиця worker призначена для зберігання даних методиста. За допомогою ідентифікатора user_id, таблиця пов'язана з таблицею user, зв'язком один до одного.

Таблиця 2.7 – Опис таблиці worker

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
Worker – таблиця, що вказує на дані студента	id	int	Ідентифікатор методиста
	user_id	int	Код, що пов'язує таблицю методистів з таблицею користувачів
	description	varchar	Опис методиста

Таблиця migration призначена для зберігання даних контролю версій бази даних.

Таблиця 2.8 – Опис таблиці migration

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
migration – таблиця, що зберігає дані про оновлення структури бази даних	version	varchar	Код, що вказує на версію бази даних
	apply_time	int	Дата, що вказує на час створення версії бази даних

Таблиця group призначена для зберігання даних груп, у яких навчаються студенти. За допомогою ідентифікатора department_id, таблиця пов'язана з таблицею department, зв'язком один до багатьох.

Таблиця 2.9 – Опис таблиці group

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
group – таблиця, що описує список груп	id	int	Ідентифікатор групи
	department_id	int	Ідентифікатор, що пов'язує таблицю груп з таблицею кафедр
	name	varchar	Назва кафедри

Таблиця faculty призначена для зберігання даних факультету.

Таблиця 2.10 – Опис таблиці faculty

					КПІ.ІП-5105.045440.01.81	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
faculty – таблиця, що описує факультети	id	int	Ідентифікатор групи
	name	varchar	Назва факультету

Таблиця diploma призначена для зберігання тематики дипломних проектів, що створюють викладачі. За допомогою ідентифікатора user_id, таблиця пов'язана з таблицею user, зв'язком один до одного. За допомогою ідентифікатора worker_id, таблиця пов'язана з таблицею worker, зв'язком один до одного.

Таблиця 2.11 – Опис таблиці diploma

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
diploma – таблиця, що описує теми диплому	id	int	Ідентифікатор теми диплому
	user_id	int	Ідентифікатор, що пов'язує таблицю користувачів з таблицею тем дипломів.
	worker_id	int	Ідентифікатор, що пов'язує таблицю методистів з таблицею тем дипломів.
	title	varchar	Тема диплому

Продовження таблиці 2.11

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
diploma – таблиця, що описує теми диплому	skills	varchar	Необхідні навички
	description	text	Опис теми диплому
	created_at	date	Дата створення теми диплому
	updated_at	date	Дата останнього оновлення теми диплому
	type	int	Ідентифікатор, що вказує на тип теми диплома
	status	int	Ідентифікатор, що вказує на статус теми диплома
	img	varchar	Посилання на малюнок теми диплома

Таблиця dialog призначена для зберігання запитів на утвердження тематики дипломних проектів від студентів до викладачів. За допомогою ідентифікатора user_id, таблиця пов'язана з таблицею user, зв'язком один до одного. За допомогою ідентифікатора diploma_id, таблиця пов'язана з таблицею diploma, зв'язком один до одного.

Таблиця 2.12 – Опис таблиці dialog

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
dialog – таблиця запитів для утвердження тем дипломних проектів	id	int	Ідентифікатор запиту
	diploma_id	int	Ідентифікатор, що пов'язує таблицю запитів з таблицею тем дипломів.
	user_id	int	Ідентифікатор, що пов'язує таблицю запитів з таблицею користувачів.
	status	int	Ідентифікатор, що позначає на статус запита
	created_at	date	Дата створення запиту
	updated_at	date	Дата останнього оновлення запиту

Таблиця department призначена для зберігання даних кафедр. За допомогою ідентифікатора faculty_id, таблиця пов'язана з таблицею faculty, зв'язком один до багатьох.

Таблиця 2.13 – Опис таблиці department

Назва таблиці	Назва стовпця	Тип даних	Детальна інформація
departament – таблиця, що описує кафедри	id	int	Ідентифікатор кафедри
	faculty_id	int	Ідентифікатор, що пов'язує таблицю кафедр з таблицею факультетів.
	name	varchar	Назва кафедри

Повна схема Баз даних застосунку наведена в графічних матеріалах.

2.5 Аналіз безпеки даних

На сьогодні проблема з безпекою даних займає багато часу і грошей. Коже сервіс і сайт, який був взламаний розкриває доступ до даних користувачів, тим самим даючи доступ злоумисникам до інших сайтів.

У веб-розробці безпека даних – це першочергова задача. Щоб повністю захистити проект від слабких місць, потрібно використовувати лише загальноприйняті технології та алгоритми. Будь-яке свавілля може призвести до втрати користувачами сайту особистих даних.

Отже потрібно зберігати пароль, використовуючи bcrypt. На сьогодні це один з небагатьох алгоритмів, який можна використовувати для зберігання паролю. Також, токен, що використовується для ідентифікації користувача, повинен бути створений захищеним генератором випадкових значень. Всі форми повинні мати csrf токен.

2.6 Висновки по розділу

У цьому розділі було розглянуто архітектуру системи, зокрема було обґрунтоване використання фреймворку yii2 та мови програмування php на

серверній частині сайту. Також були розглянуті технології, які використовуються на клієнтській частині сайту, а саме html, css, javascript, jquery.

Були описані класи та їх методи. Був проведений аналіз безпеки даних.

					КПІ.ІП-5105.045440.01.81	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

3 АНАЛІЗ ЯКОСТІ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Аналіз якості та тестування програмного забезпечення знаходиться в додатку
Аналіз якості та тестування програмного забезпечення.

					КПІ.ІП-5105.045440.01.81	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

4 ВПРОВАДЖЕННЯ ТА СУПРОВІД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Розгортання програмного забезпечення

Для розгортання проекту потрібно:

- Встановити Apache;
- Встановити Php 7.2;
- Встановити субд Mysql;
- Створити базу даних;
- Задати необхідні налаштування підключення до бази даних.

4.1.1 Встановлення Apache

Веб-сервер Apache потрібно завантажити з офіційного сайту розробника - www.apache.org.

4.1.2 Встановити Php 7.2

Php встановлюється разом з будь яким інструментом для запуску серверу, наприклад - ххамр.

4.1.3 Встановити субд Mysql

Субд Mysql встановлюється разом з ххамр

4.1.4 Створити базу даних

Створити базу даних можна, виконавши файл created.sql у корні проекту.

4.1.5 Задати необхідні налаштування підключення до бази даних

У файлі за шляхом config/config.php потрібно задати параметри підключення до створенної бази даних

4.2 Інструкція користувача

Інструкція наведена в додатку Інструкція користувача

4.3 Інструкція адміністратора

Інструкція наведена в додатку Інструкція системного програміста

					КПІ.ІП-5105.045440.01.81	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		59

ВИСНОВКИ

Отже програмний продукт був успішно розроблений. Під час кожного етапу розробки системи були описані окремі модулі розробки.

Був проведений аналіз предметної області, встановлені конкуренти та сервіси зі схожим функціоналом, були виявлені відмінності та продумані покращення до програмного продукту.

Було проведене повне моделювання застосунку, створенні функціональні та не функціональні вимоги. Для успішної реалізації бізнес процесів були створенні BPMN – діаграми, що описують головні алгоритми виконання бізнес процесів. Була розроблена архітектура системи та описані обрані технології для її розробки.

Був створений та описаний тест - план, проходження якого свідчить про успішне виконання дипломного проекту.

Також був описаний процес впровадження та супровіду програмного забезпечення, а саме: описані кроки та порядок устанавлення всіх необхідних технологій для запуску сервера. Були розроблені інструкції користувача, а саме інструкція для студента, інструкція для викладача, інструкція для методиста. Також, для адміністратора, була розроблена інструкція для адміністрування програмного продукту.

Під час розроблення дипломного проекту були знадобились усі навички здобуті під час навчання. Був опанований php – фреймворк, який розробляється в Україні.

					КПІ.ІП-5105.045440.01.81	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		60

ПЕРЕЛІК ПОСИЛАНЬ

- 1) <https://uk.wikipedia.org/wiki/PHP> - PHP
- 2) <https://www.yiiframework.com/doc/guide/2.0/uk> - Повний посібник з Yii 2.0
- 3) <https://forum.yiiframework.com/> - форум yii framework
- 4) <https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C-%D0%B2%D0%B8%D0%B4-%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D0%B5%D1%80> – модель – вид – контролер
- 5) <https://jquery.com/> - jquery
- 6) <https://uk.wikipedia.org/wiki/%D0%92%D0%B5%D0%B1-%D1%80%D0%BE%D0%B7%D1%80%D0%BE%D0%B1%D0%BA%D0%B0> – веб – розробка
- 7) <https://php.ru/forum/> - форум програмістів
- 8) Робін Ніксон (2010). Learning PHP, MySQL & JavaScript: With JQuery, CSS & HTML5 с. 55
- 9) Пол Дюбуа (2002) MySQL Cookbook с. 23

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Web – застосування для узгодження тематики дипломних проектів

Технічне завдання

КПІ.ІП-5105.045440-02-91

“ПОГОДЖЕНО”

Керівник проекту:

_____ О.Д. Фіногенов

Нормоконтроль:

_____ М.М. Головченко

Виконавець:

_____ Р.С. Гохгалтер

Київ – 2019 року

ЗМІСТ

1	НАЙМЕНУВАННЯ ТА ГАЛУЗЬ ЗАСТОСУВАННЯ.....	4
2	ПІДСТАВА ДЛЯ РОЗРОБКИ.....	5
3	ПРИЗНАЧЕННЯ РОЗРОБКИ	6
4	ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	7
4.1	ВИМОГИ ДО ФУНКЦІОНАЛЬНИХ ХАРАКТЕРИСТИК.....	7
4.1.1	<i>Програмне забезпечення повинно забезпечувати виконання наступних основних функцій:.....</i>	<i>7</i>
4.1.2	<i>Розробку виконати на платформі windows 10.....</i>	<i>7</i>
4.1.3	<i>Додаткові вимоги:.....</i>	<i>7</i>
4.2	ВИМОГИ ДО НАДІЙНОСТІ	8
4.2.1	<i>Передбачити контроль введення інформації.....</i>	<i>8</i>
4.2.2	<i>Передбачити захист від некоректних дій користувача.....</i>	<i>8</i>
4.2.3	<i>Забезпечити цілісність інформації в базі даних.....</i>	<i>8</i>
4.3	УМОВИ ЕКСПЛУАТАЦІЇ	8
4.4	ВИМОГИ ДО СКЛАДУ І ПАРАМЕТРІВ ТЕХНІЧНИХ ЗАСОБІВ	8
4.4.1	<i>Програмне забезпечення повинно функціонувати на IBM-сумісних персональних комп'ютерах.....</i>	<i>8</i>
4.4.2	<i>Мінімальна конфігурація технічних засобів:</i>	<i>8</i>
4.5	ВИМОГИ ДО ІНФОРМАЦІЙНОЇ ТА ПРОГРАМНОЇ СУМІСНОСТІ	8
4.5.1	<i>Програмне забезпечення повинно працювати під управлінням операційних систем сімейства WIN32 (Windows'XP, Windows NT і т.д.) або Unix.....</i>	<i>8</i>
4.5.2	<i>Програмне забезпечення повинно працювати під операційною системою сімейства windows та linux.....</i>	<i>8</i>
4.6	ВИМОГИ ДО МАРКУВАННЯ ТА ПАКУВАННЯ.....	8
4.7	ВИМОГИ ДО ТРАНСПОРТУВАННЯ ТА ЗБЕРІГАННЯ	8
4.8	СПЕЦІАЛЬНІ ВИМОГИ.....	8
5	ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ	9
5.1	ПРОГРАМНІ МОДУЛІ, КОТРІ РОЗРОБЛЯЮТЬСЯ, ПОВИННІ БУТИ ЗАДОКУМЕНТОВАНІ, ТОБТО ТЕКСТИ ПРОГРАМ ПОВИННІ МІСТИТИ ВСІ НЕОБХІДНІ КОМЕНТАРІ.....	9
5.2	ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ПОВИННО МАТИ ДОВІДНИКОВУ СИСТЕМУ	9

5.3 У склад супроводжувальної документації повинні входити наступні документи: 9

5.3.1 Пояснювальна записка не менше ніж на 60 аркушах формату А4 (без додатків 5.3.2 - 5.3.6).....	9
5.3.2 Технічне завдання.	9
5.3.3 Опис програми.	9
5.3.4 Керівництво користувача.	9
5.3.5 Керівництво адміністратора.....	9
5.3.6 Програма та методика тестування	9

5.4 Графічна частина повинна бути виконана у форматі А3, котрі включаються у якості додатків до пояснювальної записки:9

5.4.1 Схема структурна варіантів використання	9
5.4.2 Схема структурна класів програмного забезпечення	9
5.4.3 Схема бази даних.....	9

6 СТАДІЇ І ЕТАПИ РОЗРОБКИ.....10

7 ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ.....12

7.1 Види випробувань	12
----------------------------	----

1 НАЙМЕНУВАННЯ ТА ГАЛУЗЬ ЗАСТОСУВАННЯ

Назва розробки: web – застосування для узгодження тематики дипломних проектів

Галузь застосування:

Наведене технічне завдання поширюється на розробку Web-додатку, клієнту для доступу до Internet-ресурсів [045440], котра використовується для утвердження тем дипломних проектів та призначена для освітньої галузі для використання студентами, викладачами, методистами.

					КПІ.ІП-5105.045440.02.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		4

2 ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки Web – застосування для узгодження тематики дипломних проєктів є завдання на дипломне проєктування, затверджене кафедрою автоматизованих систем обробки інформації і управління Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» (КПІ ім.Ігоря Сікорського).

					КПІ.ІП-5105.045440.02.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		5

3 ПРИЗНАЧЕННЯ РОЗРОБКИ

Розробка призначена для студентів, які не можуть знайти теми дипломного проекту, викладачам, якій мають теми дипломних проектів та методистам, яким потрібно автоматизувати процеси.

Метою розробки є спрощення утвердження та пошуку дипломних проектів студентам, створення тем дипломних проектів на основі задач, які повинні допомогти автоматизувати роботу з інформацією в університеті, шляхом створення web – додатку, який має функціонал для розміщення та затвердження тем дипломних проектів.

					КПІ.ІП-5105.045440.02.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		6

4 ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Вимоги до функціональних характеристик

4.1.1 Програмне забезпечення повинно забезпечувати виконання наступних основних функцій:

4.1.1.1 Для користувача:

- створення, редагування, видалення теми дипломного проекту;
- створення задачі від методиста;
- затвердження теми дипломного проекту;
- авторизація;
- вихід з облікового запису;
- реєстрація.

4.1.1.2 Для адміністратора системи:

- авторизація;
- вихід з облікового запису;
- управління користувачами.

4.1.2 Розробку виконати на платформі windows 10

4.1.3 Додаткові вимоги:

- Сайт повинен буди на українській або англійській мові;

4.2 Вимоги до надійності

4.2.1 Передбачити контроль введення інформації.

4.2.2 Передбачити захист від некоректних дій користувача.

4.2.3 Забезпечити цілісність інформації в базі даних.

4.3 Умови експлуатації

4.4 Вимоги до складу і параметрів технічних засобів

4.4.1 Програмне забезпечення повинно функціонувати на IBM-сумісних персональних комп'ютерах.

4.4.2 Мінімальна конфігурація технічних засобів:

Тип процесору intel i2.

Об'єм ОЗП 2 гб.

Операційна система Windows, або Linux.

4.5 Вимоги до інформаційної та програмної сумісності

4.5.1 Програмне забезпечення повинно працювати під управлінням операційних систем сімейства WIN32 (Windows'XP, Windows NT і т.д.) або Unix.

4.5.2 Програмне забезпечення повинно працювати під операційною системою сімейства windows та linux.

4.6 Вимоги до маркування та пакування

Вимоги до маркування та пакування не пред'являються.

4.7 Вимоги до транспортування та зберігання

Вимоги до транспортування та зберігання не пред'являються.

4.8 Спеціальні вимоги

Згенерувати установчу версію програмного забезпечення.

					КПІ.ІП-5105.045440.02.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

5 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ

5.1 Програмні модулі, котрі розробляються, повинні бути задокументовані, тобто тексти програм повинні містити всі необхідні коментарі.

5.2 Програмне забезпечення повинно мати довідникову систему

5.3 У склад супроводжувальної документації повинні входити наступні документи:

5.3.1 Пояснювальна записка не менше ніж на 60 аркушах формату А4 (без додатків 5.3.2 - 5.3.6).

5.3.2 Технічне завдання.

5.3.3 Опис програми.

5.3.4 Керівництво користувача.

5.3.5 Керівництво адміністратора

5.3.6 Програма та методика тестування

5.4 Графічна частина повинна бути виконана у форматі А3, котрі включаються у якості додатків до пояснювальної записки:

5.4.1 Схема структурна варіантів використання

5.4.2 Схема структурна класів програмного забезпечення

5.4.3 Схема бази даних

6 СТАДІЇ І ЕТАПИ РОЗРОБКИ

	Назва етапу	Строк,	Звітність
1	Вивчення літератури за тематикою проекту	20.04.19	
2	Розробка технічного завдання	24.04.19	Технічне завдання
3	Аналіз вимог та уточнення специфікацій	25.04.19	Специфікації програмного забезпечення
4	Проектування структури програмного забезпечення, проектування компонентів	29.04.19	Схема структурна програмного забезпечення та специфікація компонентів (діаграма класів, схема алгоритму ...)
5	Програмна реалізація програмного забезпечення	16.05.19	Тексти програмного забезпечення
6	Тестування програмного забезпечення	20.05.19	Тести, результати тестування
7	Розробка матеріалів текстової частини проекту	26.05.19	Пояснювальна записка.
8	Розробка матеріалів графічної частини проекту	07.06.19	Графічний матеріал проекту

КПІ.ІП-5105.045440.02.91

9	Оформлення технічної документації проекту	09.06.19	Технічна документація
---	---	----------	--------------------------

					КПІ.ІП-5105.045440.02.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

7 ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ

7.1 Види випробувань

Тестування розробленого програмного продукту виконується відповідно до “Програми та методики тестування”.

					КПІ.ІП-5105.045440.02.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Web – застосування для узгодження тематики дипломних проектів

Програма та методика тестування

КПІ.ІІ-5105.045430.03.51

“ПОГОДЖЕНО”

Керівник проекту:

_____ О.Д. Фіногенов

Нормоконтроль:

_____ М.М. Головченко

Виконавець:

_____ Р.С. Гохгалтер

Київ – 2019 року

ЗМІСТ

1	АНАЛІЗ ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	3
2	ПІДХОДИ ДО ТЕСТУВАННЯ	5
2.1	КОМПОНЕНТНЕ ТЕСТУВАННЯ	5
2.2	ІНТЕГРАЦІЙНЕ ТЕСТУВАННЯ	5
2.3	ТЕСТУВАННЯ ПРОДУКТИВНОСТІ	5
2.4	ТЕСТУВАННЯ ШВИДКОДІЇ	5
3	КРИТЕРІЇ ПРОХОДЖЕННЯ ТЕСТУВАННЯ	7
3.1	КОМПОНЕНТНЕ ТЕСТУВАННЯ	7
3.2	ІНТЕГРАЦІЙНЕ ТЕСТУВАННЯ	7
3.3	ТЕСТУВАННЯ ШВИДКОДІЇ	7
3.4	ТЕСТУВАННЯ ПРОДУКТИВНОСТІ	7
4	ПРОЦЕС ТЕСТУВАННЯ	8
4.1	ДАНІ ДО ТЕСТІВ	8
4.2	ЗАДАЧІ ТЕСТУ	8
4.3	ПЛАН ВИКОНАННЯ	8
5	ВИМОГИ ДО СЕРЕДОВИЩА	9
5.1	АПАРАТНА ЧАСТИНА	9
5.2	ПРОГРАМНА ЧАСТИНА	9
5.3	ВИМОГИ ДО БЕЗПЕКИ	9
5.3.1	Інструменти	9
6	ОПИС КОНТРОЛЬНОГО ПРИКЛАДУ	10

1 АНАЛІЗ ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

На сьогодні дуже багато компаній, навіть, маючи величезний матеріальний і людський ресурс, через неякісний код і архітектуру, впираються у постійно зростаючу складність розширення програмний продуктів, через складність тестування нових функцій. Тож тестування програмного коду має велике значення.

Тестування повинно підтвердити необхідний рівень якості, щоб програмний продукт можна було назвати виконаним.

Через велику кількість різноманітних програмних рішень, не існує універсальних методологій при створення плану тестування. Знаходження помилок це в деякій мірі творча робота, завжди потрібно виконувати якісь неочевидні дії.

Однім з плюсів використання фреймворку yii2 є наявність режиму тестування, який веде записи усіх програмних помилок, що сталися під час роботи із системою.

Отже тестуванню підлягають наступні функції та методи:

- реєстрація викладача, студента, методиста;
- авторизація викладача, студента, методиста;
- створення теми дипломного проекту;
- створення завдання методистом;
- створення теми дипломного проекту на основі завдання методиста;
- створення запиту на виконання теми дипломного проекту;
- підтвердження запиту на виконання теми дипломного проекту;
- блокування користувача адміністратором;
- підтвердження викладача, або методиста.

Опрацьовані наступні тестові модулі:

- реєстрація викладача, студента, методиста;
- авторизація викладача, студента, методиста;

					КПІ.ІП-5201.045430.03.51	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		3

- створення теми дипломного проекту;
- створення завдання методистом;
- створення теми дипломного проекту на основі завдання методиста;
- створення запиту на виконання теми дипломного проекту;
- підтвердження запиту на виконання теми дипломного проекту;
- блокування користувача адміністратором;
- підтвердження викладача, або методиста.

					КПІ.ІП-5201.045430.03.51	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		4

2 ПІДХОДИ ДО ТЕСТУВАННЯ

Будуть використані такі підходи до тестування:

- інтеграційний підход;
- компонентний підход;
- підход продуктивності;
- підход швидкодії.

2.1 Компонентне тестування

Методом компонентного тестування будуть перевірені логічно окремі частини веб сервісу, такі як:

- методи для роботи з базою даних;
- допоміжні функції.

2.2 Інтеграційне тестування

Методом інтеграційного тестування буде перевірена взаємодія клієнтської частини з серверною частиною

2.3 Тестування продуктивності

Методом тестування продуктивності буде перевірена швидкодія виконання запитів до бази даних.

2.4 Тестування швидкодії

Тестування швидкодії виконується паралельно з виконанням усіх інших тестів. Після будь-яких дій, у режимі розробника фреймворка уіі2 записую результати та час виконання усіх запитів до бази даних, а також час виконання запиту до самого веб-серверу.

					КПІ.ІП-5201.045430.03.51	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		5

Фреймоврк уіі2, має розширення, яке самостійно вказує на запити, на серверну частину, які виконались з помилками, або виконувались занадто довго.

					КПІ.ІП-5201.045430.03.51	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		6

3 КРИТЕРІЇ ПРОХОДЖЕННЯ ТЕСТУВАННЯ

3.1 Компонентне тестування

Для компонентного тестування критерієм проходження є успішне виконання кожного запиту до бази даних та успішне виконання усіх методів обробки даних на сервері.

3.2 Інтеграційне тестування

Для інтеграційного тестування критерієм проходження є успішне виконання усіх запитів, що надходять з клієнтської частини системи до серверної частини системи.

3.3 Тестування швидкодії

Для тестування швидкодії критерієм проходження є час виконання кожного запиту до бази даних, який задовольняє загальним критерієм швидкості роботи запитів.

3.4 Тестування продуктивності

Для тестування продуктивності критерієм проходження є виконання основних функцій системи за той час виконання, який загальним критерієм швидкості роботи сайтів. Окрім цього тестування вважається неуспішним, якщо хоча б одна функція системи не була виконана.

4 ПРОЦЕС ТЕСТУВАННЯ

4.1 Дані до тестів

Вхідними даними для інтеграційного тестування є запити від клієнтської частини системи до серверної частини системи. Вихідними даними цього виду тестування є відповідь від серверної частини програми.

Вхідними даними для компонентного тестування є ті дані, які будуть змінені під час виконання дій системи та повернені, як вихідні дані.

Вхідними даними для тестування швидкості є дані, які використовуються для виконання запитів до бази даних. Вихідними даними цього виду тестування є результатом виконання запиту до бази даних.

Вхідними даними для тестування продуктивності є усі вхідні дані системи, які викликають взаємодію між клієнтською частиною системи і серверною частиною системи.

4.2 Задачі тесту

Задачею тесту є перевірка готовності програмного продукту до випуску продукту на ринок.

4.3 План виконання

Тестування виконується у такому порядку:

- компонентне тестування;
- інтеграційне тестування;
- тестування швидкодії;
- тестування продуктивності.

5 ВИМОГИ ДО СЕРЕДОВИЩА

5.1 Апаратна частина

Вимоги до апаратної частини співпадають з вимогами з технічного завдання.

5.2 Програмна частина

Для виконання тестування платформа повинна мати операційну систему на базі Linux, Windows.

5.3 Вимоги до безпеки

Тестування повинне проводитись на локальному сервері, або закритому онлайн-сервері.

5.3.1 Інструменти

Для виконання тестування використовувати наступні програмні інструменти:

- Yii2.

6 ОПИС КОНТРОЛЬНОГО ПРИКЛАДУ

У якості контрольного прикладу буде розглянуте тестування авторизації користувача з використанням фреймворку yii2.

Вхідні дані:

- вірна електронна пошта користувача та вірний пароль;
- не вірна електронна пошта користувача та вірний пароль;
- вірна електронна пошта користувача та не вірний пароль;

Вихідні дані:

- повідомлення про успішну авторизація;
- повідомлення про не вірну електронну пошту, або пароль;
- повідомлення про не вірну електронну пошту, або пароль.

Отже, тестування показало, що система врно реагує на всі можливі варіанти подій при авторизації користувача. Фреймворк у всіх випадках показав вірне виконання скриптів на сервері та не вказав на довге виконання запитів - рисунок 3.1.

Time	Memory	DB	Events	Asset Bundles	User
01:08:48.065					
01:08:48.065					
01:08:48.074					
01:08:48.083					
01:08:48.083					
01:08:48.106					
01:08:48.109					

Рисунок 3.1 - Результат авторизації

Отже, тест успішно виконан.

КПІ.ІП-5201.045430.03.51

					КПІ.ІП-5201.045430.03.51	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Web – застосування для узгодження тематики дипломних проектів

Керівництво користувача

КПІ.ІІ-5105.045440.06.33

“ПОГОДЖЕНО”

Керівник проекту:

_____ О.Д. Фіногенов

Нормоконтроль:

_____ М.М. Головченко

Виконавець:

_____ Р.С. Гохгалтер

Київ – 2019 року

ЗМІСТ

1	КЕРІВНИЦТВО КОРИСТУВАЧА	3
1.1	ПЕРЕГЛЯД ТЕМ ДИПЛОМНИХ РОБІТ	3
1.2	РЕЄСТРАЦІЯ КОРИСТУВАЧА	4
1.2.1	Реєстрація для студента	5
1.2.2	Реєстрація для викладача	7
1.2.3	Реєстрація для методиста	8
1.3	ВХІД КОРИСТУВАЧА В ОБЛКОВИЙ ЗАПИС	9
2	КЕРІВНИЦТВО СТУДЕНТА	10
2.1	СТВОРЕННЯ ЗАПИТУ НА УТВЕРДЖЕННЯ ТЕМАТИКИ ДИПЛОМНОГО ПРОЕКТУ	10
2.2	ПЕРЕГЛЯД СТАНУ ЗАПИТУ НА УТВЕРДЖЕННЯ ТЕМАТИКИ ДИПЛОМНОГО ПРОЕКТ....	12
3	КЕРІВНИЦТВО ВИКЛАДАЧА.....	13
3.1	СТВОРЕННЯ, РЕДАГУВАННЯ, ВИДАЛЕННЯ ТЕМИ ДИПЛОМНОГО ПРОЕКТУ	13
3.2	СТВОРЕННЯ, РЕДАГУВАННЯ, ВИДАЛЕННЯ ТЕМИ ДИПЛОМНОГО ПРОЕКТУ НА ОСНОВІ ЗАДАЧІ ВІД МЕТОДИСТА	16
3.3	ПЕРЕГЛЯД ВЛАСНИХ ТЕМ ДИПЛОМНИХ ПРОЕКТІВ	19
3.4	УПРАВЛІННЯ ЗАПИТАМИ НА УТВЕРДЖЕННЯ ТЕМИ ДИПЛОМНОГО ПРОЕКТУ	19
3.4.1	Відхилення запиту на утвердження дипломної тематики	20
3.4.2	Утвердження теми дипломного проекту	21
4	КЕРІВНИЦТВО МЕТОДИСТА	23
4.1	СТВОРЕННЯ ЗАДАЧІ.....	23
4.2	РЕДАГУВАННЯ ЗАДАЧІ	24
4.3	ВИДАЛЕННЯ ЗАДАЧІ	26
4.4	ПЕРЕГЛЯД СТАТУСУ ЗАПИТІВ	26

1 КЕРІВНИЦТВО КОРИСТУВАЧА

Користувач може авторизуватись, зареєструватись та передивитись теми дипломних робіт.

Роль користувача розділяється на 3 актори:

- студент;
- викладач;
- методист.

Щоб отримати доступ до функцій одного з акторів, користувач повинен зареєструватися, обравши реєстрацію для одного з акторів.

1.1 Перегляд тем дипломних робіт

Спочатку потрібно відкрити будь-який браузер та перейти за посиланням у сайт, рисунок 1.1.

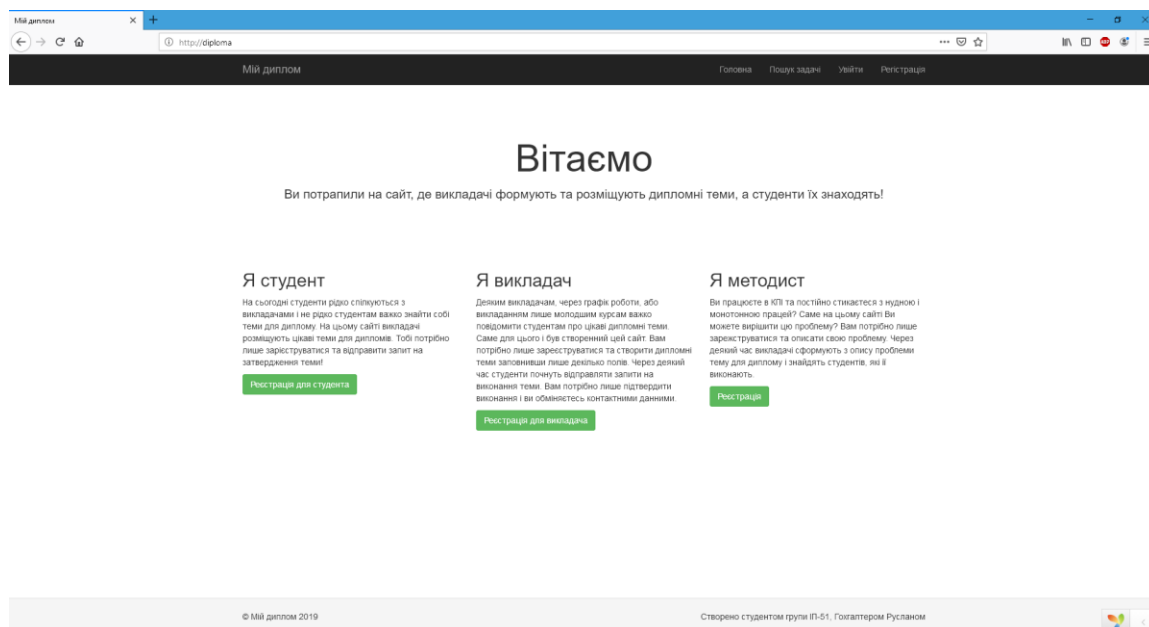


Рисунок 1.1 – Сайт

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		3

Далі потрібно натиснути на пункт меню «пошук задач». Відкриється сторінка із списком тем дипломних робіт – рисунок 1.2.

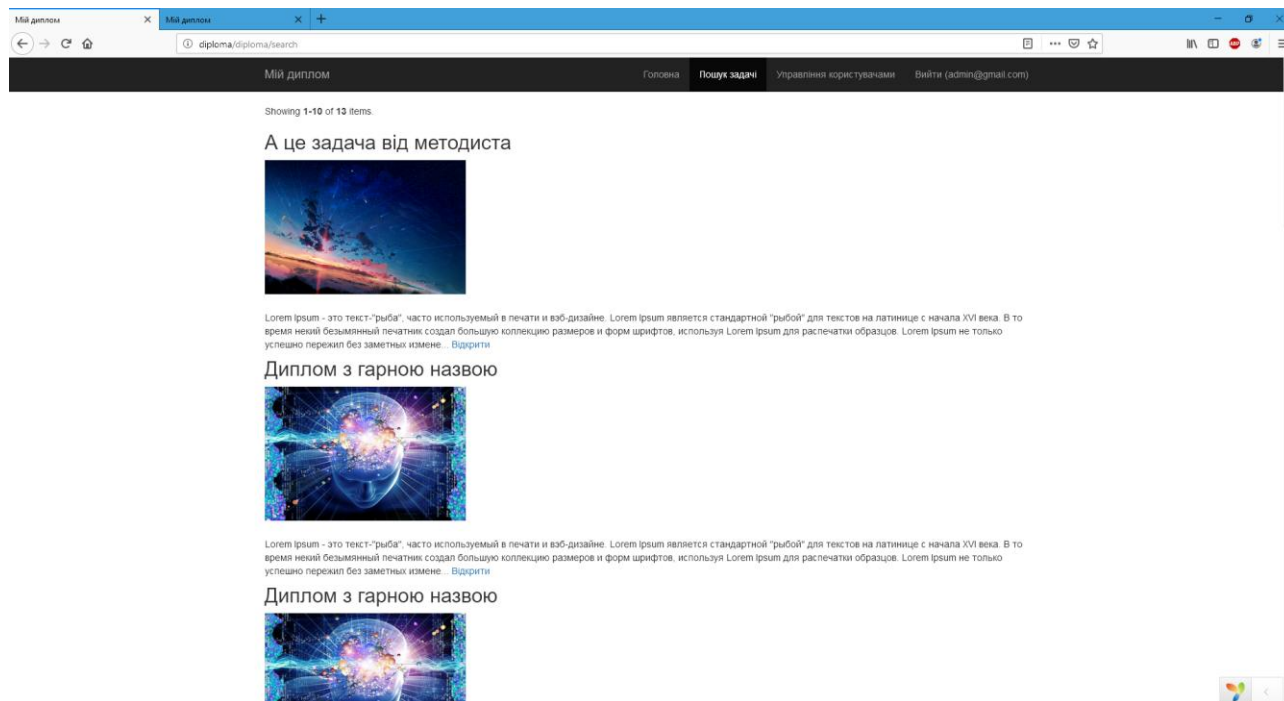


Рисунок 1.2 – Список тем дипломних робіт

1.2 Реєстрація користувача

Щоб зареєструватись, користувач повинен натиснути на кнопку «Реєстрація». Далі відкриється сторінка, яка запропонує реєстрацію для кожного з типів акторів – рисунок 1.3.

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		4

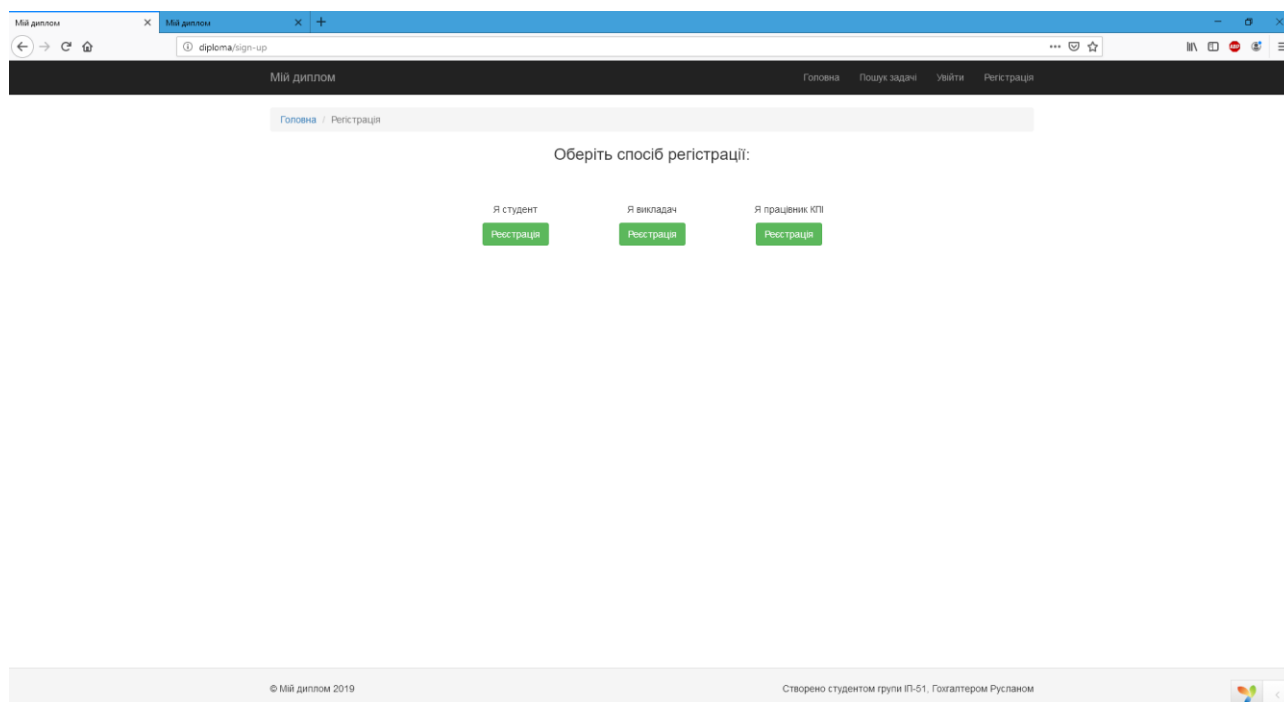


Рисунок 1.3 – Вибір реєстрації користувача

1.2.1 Реєстрація для студента

Для реєстрації студентом, користувач повинен натиснути на кнопку «Реєстрація» під текстом «Я студент». Відкриється сторінка реєстрації для студента – рисунок 1.4

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		5

The screenshot shows a web browser window with the address bar displaying 'diploma/sign-up/student'. The page title is 'Мій диплом'. The navigation bar includes links: 'Головна', 'Пошук задань', 'Вийти', and 'Регістрація'. The main content area is titled 'Реєстрація студента' and contains the instruction 'Введіть дані для реєстрації:'. The form includes input fields for 'Ім'я', 'Прізвище', 'Електронна адреса', and 'Пароль'. Below these are radio buttons for 'Стать' (Чоловіча, Жінча), a dropdown for 'Оберіть курс' (currently showing '1 курс'), and another dropdown for 'Оберіть групу' (currently showing 'ІП-51'). A blue 'Реєстрація' button is at the bottom of the form. The footer contains copyright information '© Мій диплом 2019' and a note 'Створено студентом групи ІП-51, Гогайлером Русланом'.

Рисунок 1.4 – Реєстрація для студента

Для успішної реєстрації студент повинен заповнити форму реєстрації, зокрема вказати курс і групу навчання. Далі користувач повинен натиснути на кнопку «Реєстрація». Після натиснення на кнопку, система зареєструє користувача, як студента та автоматично здійснить вхід у обліковий запис користувача. Після цього користувач буде перенаправлений а головну сторінку сайта – рисунок 1.5.

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		6

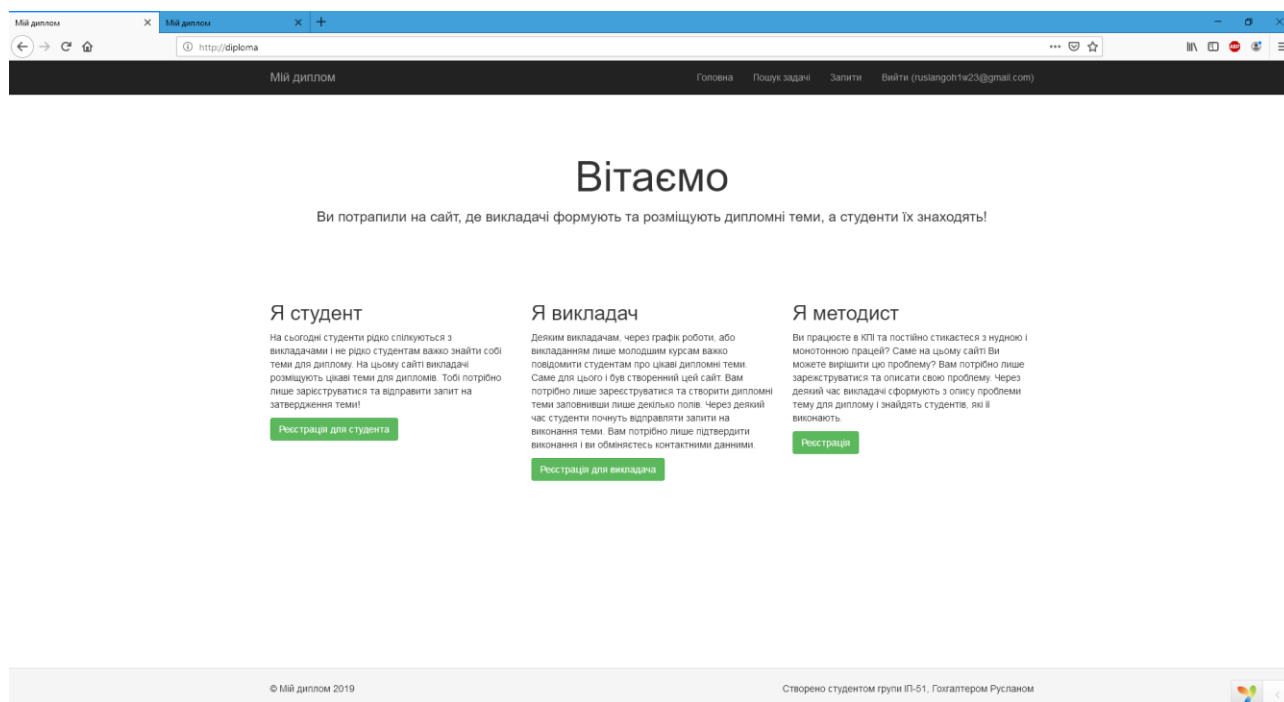


Рисунок 1.5 – Перенаправлення після реєстрації

1.2.2 Реєстрація для викладача

Для реєстрації викладача, користувач повинен натиснути на кнопку «Реєстрація» під текстом «Я викладач». Відкриється сторінка реєстрації для викладача – рисунок 1.6

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7

Мій диплом / Реєстрація студента

Реєстрація викладача

Введіть дані для реєстрації

Ім'я

Name cannot be blank.

Прізвище

Електронна адреса

Пароль

Стать
☐ Чоловіча
☐ Жіноча

Наукова ступінь

Оберіть факультет

Оберіть кафедру

Реєстрація

© Мій диплом 2019 Створено студентом групи ІП-51, Гогаптером Русланом

Рисунок 1.6 – Реєстрація для викладача

Для успішної реєстрації користувач повинен заповнити форму реєстрації, зокрема вказати факультет, кафедру викладання та науковий ступінь. Далі користувач повинен натиснути на кнопку «Реєстрація». Після натиснення на кнопку, система зареєструє користувача, як викладача та автоматично здійснить вхід у обліковий запис викладача. Після цього користувач буде перенаправлений на головну сторінку сайту.

1.2.3 Реєстрація для методиста

Для реєстрації методиста, користувач повинен натиснути на кнопку «Реєстрація» під текстом «Я методист». Відкриється сторінка реєстрації для методиста – рисунок 1.7

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

Рисунок 1.7 – Реєстрація для методиста

Для успішної реєстрації користувач повинен заповнити форму реєстрації, зокрема надати опис своєї діяльності. Далі користувач повинен натиснути на кнопку «Реєстрація». Після натиснення на кнопку, система зареєструє користувача, як методиста та автоматично здійснить вхід у обліковий запис методиста. Після цього користувач буде перенаправлений на головну сторінку сайту.

1.3 Вхід користувача в обліковий запис

Для авторизації користувач повинен натиснути на кнопку «Увійти». Користувач буде перенаправлений на сторінку авторизації. Далі користувач повинен записати у форму авторизації свою пошту та пароль та натиснути на кнопку «Авторизація». Після авторизації користувач буде перенаправлений на головну сторінку сайту

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

2 КЕРІВНИЦТВО СТУДЕНТА

Студент має всі функції не зареєстрованого користувача та такі можливості на сайті:

- створення запиту на утвердження тематики дипломного проекту;
- перегляд стану запиту на утвердження тематики дипломного проекту.

2.1 Створення запиту на утвердження тематики дипломного проекту

Щоб створити запит, студент повинен відкрити список тем дипломних проектів, натиснувши на кнопку «Пошук задачі» - рисунок 2.1 та обравши потрібну тему, натиснути на «Відкрити» пі темою – рисунок 2.2.

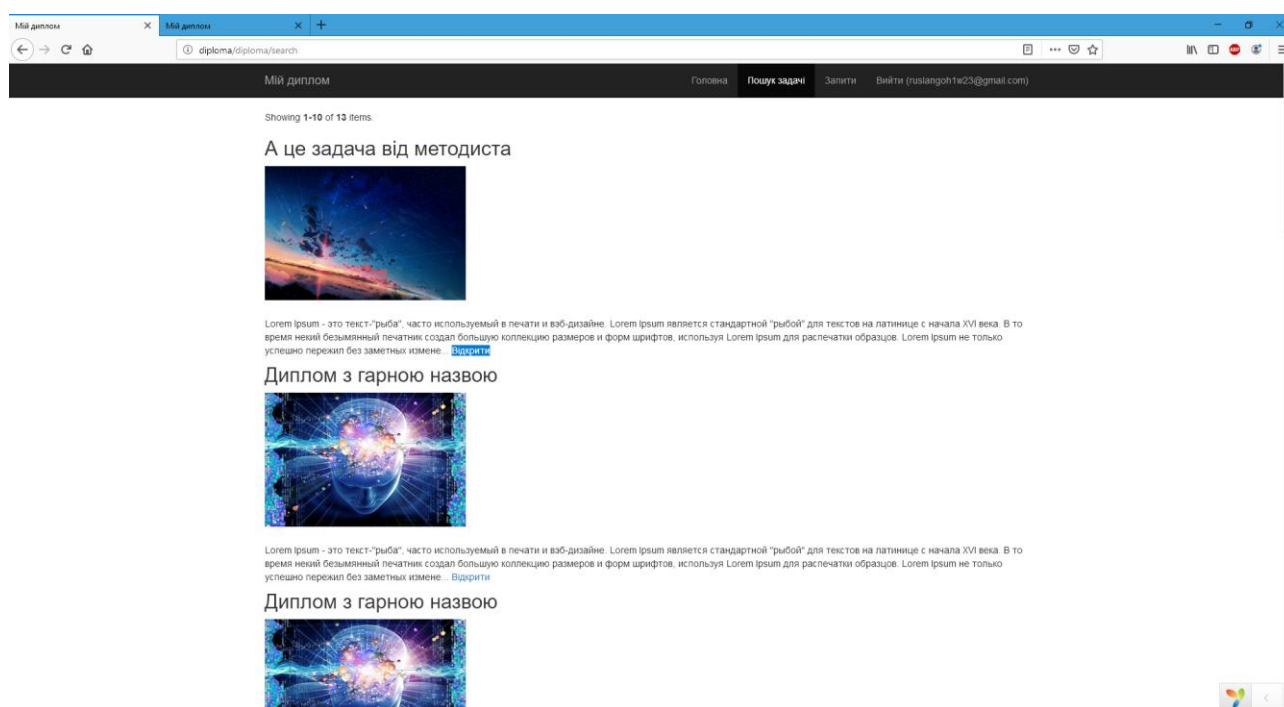


Рисунок 2.1 – Список тем дипломних проектів

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		10

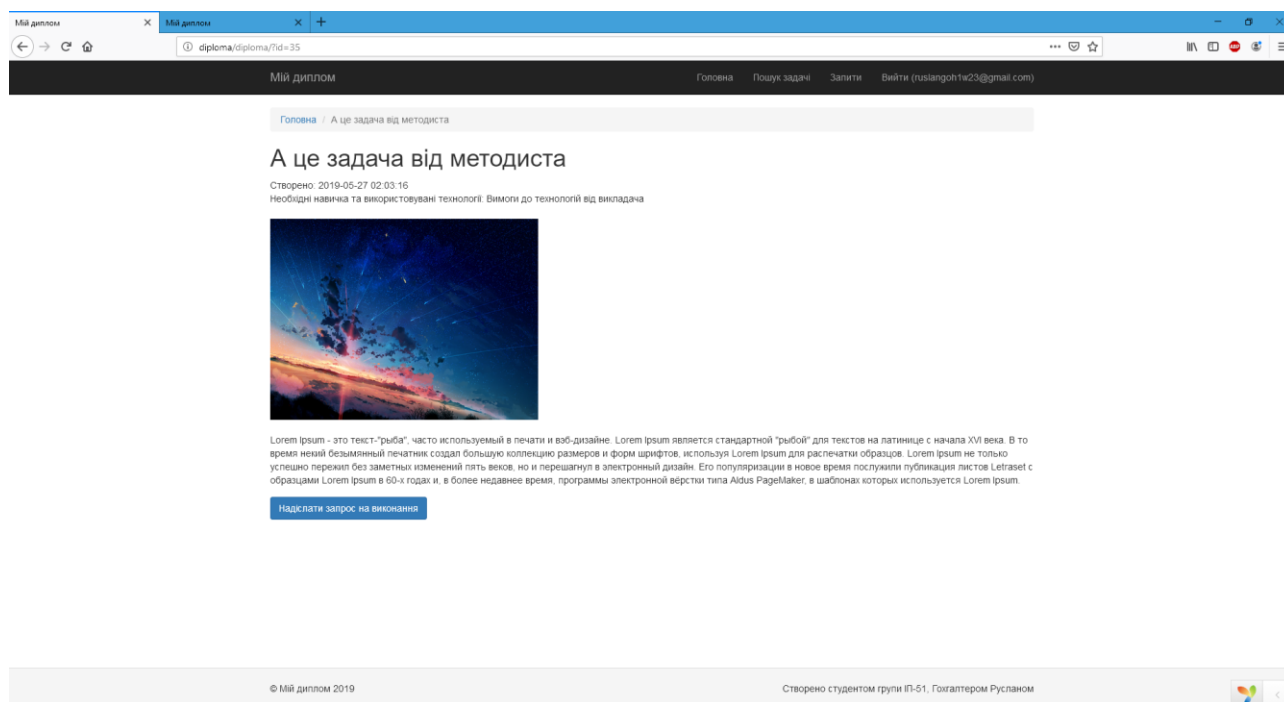


Рисунок 2.2 – Відкрита тема дипломного проекту

Натиснувши на кнопку «Надіслати запит на виконання», буде створений запит на виконання – рисунок 2.3.

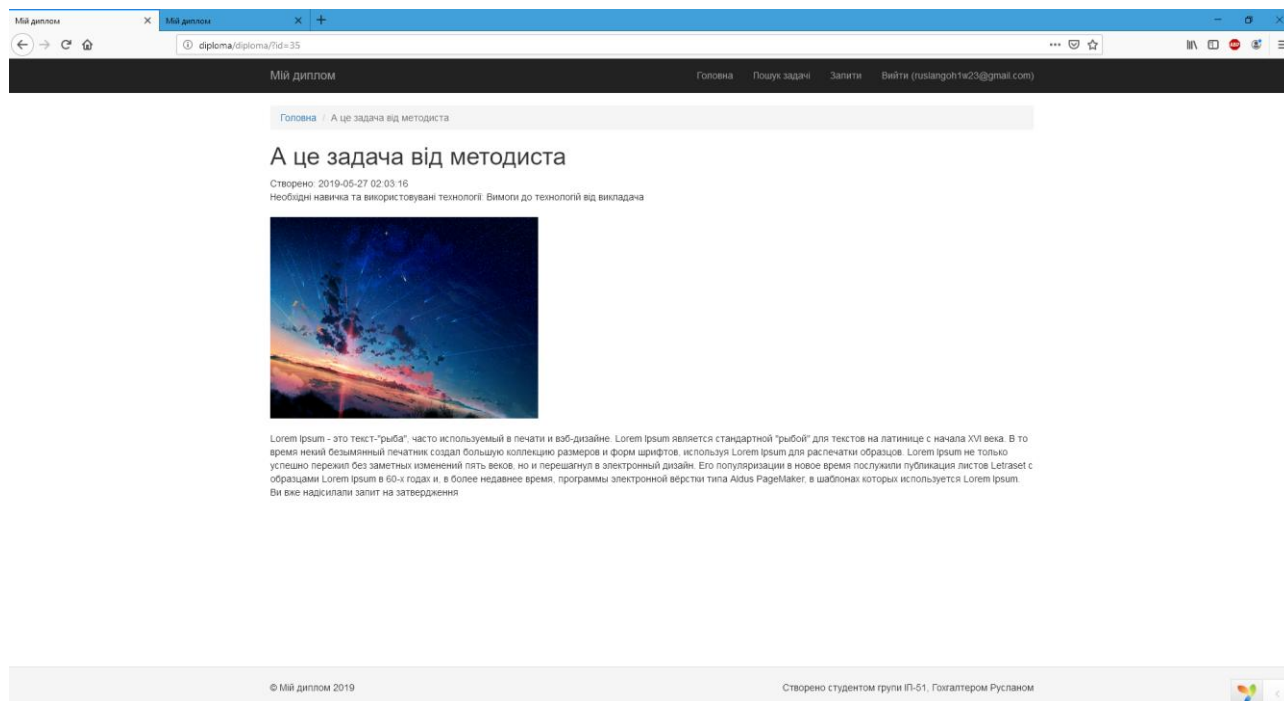


Рисунок 2.3 – Результат створенного запиту

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

2.2 Перегляд стану запиту на утвердження тематики дипломного проекту

Натиснувши на посилання «Запити», студент перейде на список зроблених запитів – рисунок 2.4.

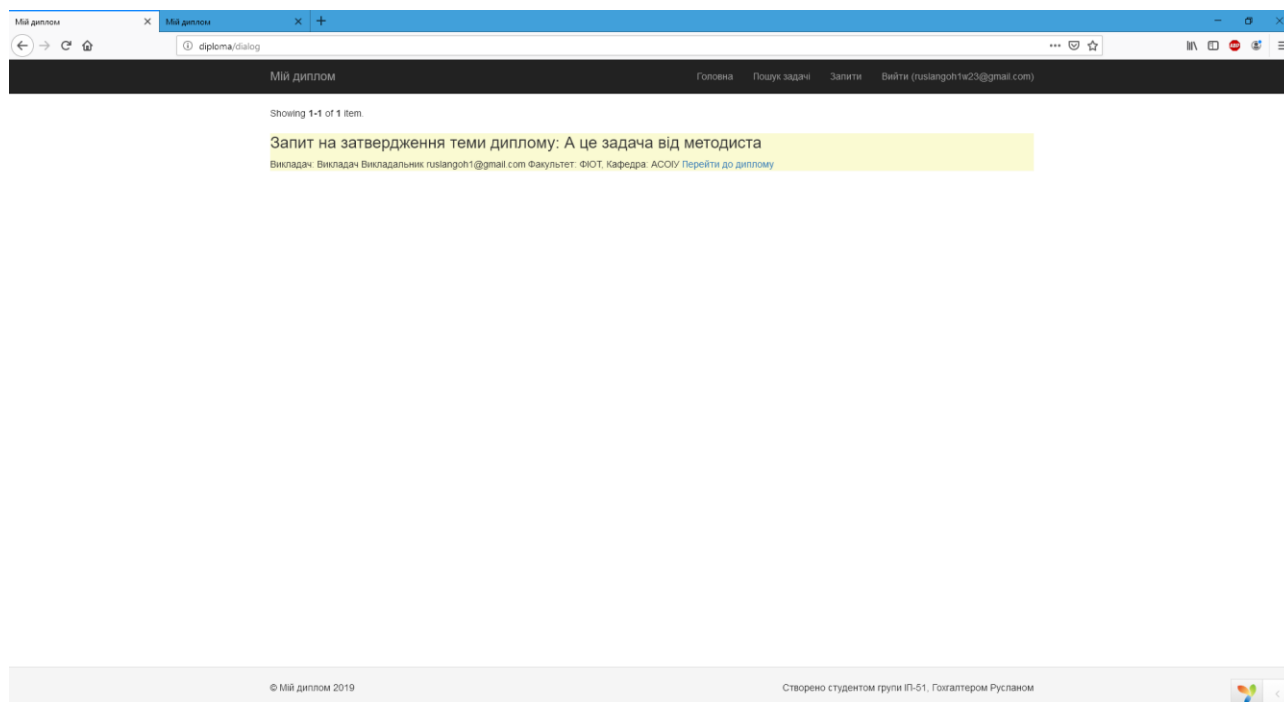


Рисунок 2.4 – Список створених запитів

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

3 КЕРІВНИЦТВО ВИКЛАДАЧА

Викладач має всі функції не зареєстрованого користувача та такі можливості на сайті:

- створення, редагування, видалення теми дипломного проекту;
- створення, редагування, видалення теми дипломного проекту на основі задачі від методиста;
- управління запитами на утвердження тематики дипломного проекту;
- перегляд своїх тем дипломних проектів;
- перегляд задач від методистів.

3.1 Створення, редагування, видалення теми дипломного проекту

Для створення теми дипломного проекту, викладач повинен натиснути на кнопку «Створити тему диплому». Після цього викладач перейде на сторінку створення теми дипломного проекту – рисунок 3.1.

The screenshot shows a web browser window with the URL 'diploma/diploma/create'. The page title is 'Мій диплом'. The navigation bar includes links: 'Головна', 'Пошук задач', 'Створити тему диплома', 'Задачі від методистів', 'Міє теми', 'Запити', and 'Вийти (ruslangonb@gmail.com)'. The main content area is titled 'Створення теми диплома' and contains the following form elements:

- A text input field for 'Назва' (Title) with a red border and a message 'Title cannot be blank'.
- A section for 'Зображення для уваги' (Image for attention) with a button 'Обзор...' and a message 'Файл не выбран'.
- A text input field for 'Опис (від 200 до 2000 символів)' (Description).
- A text input field for 'Навички та технології, що потрібно знати' (Skills and technologies to be known).
- A blue button labeled 'Створити' (Create).

At the bottom of the page, there is a footer with the text '© Мій диплом 2019' and 'Створено студентом групи ІП-51, Гогаптером Русланом'.

Рисунок 3.1 – Форма створення теми дипломного проекту

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

Після заповнення форми викладач повинен натиснути на кнопку «Створити». Після натиснення на кнопку, створиться нова тема дипломного проекту та викладач буде автоматично перенаправлений на сторінку цієї теми диплома – рисунок 3.2.

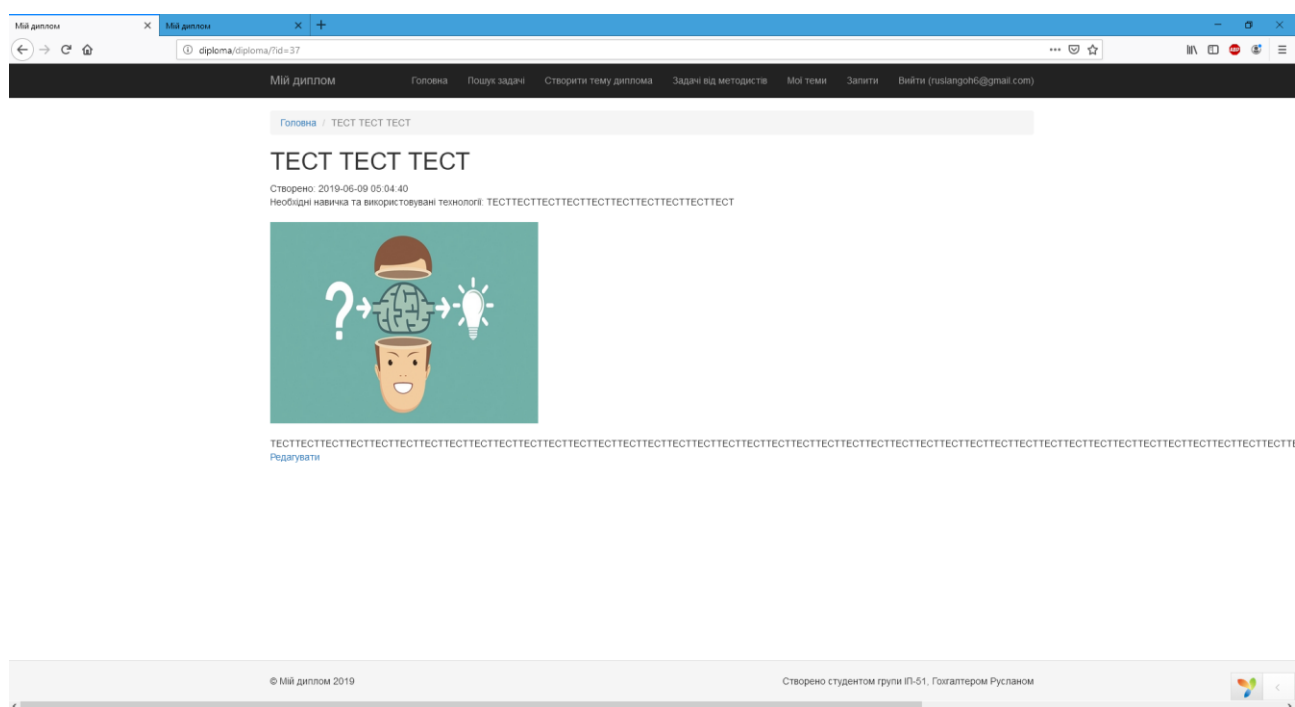


Рисунок 3.2 – Створена тема дипломного проекту

Для редагування теми дипломного проекту, викладач повинен натиснути на кнопку «Редагувати» у відкритій темі дипломного проекту. Буде відкрита форма редагування, що аналогічна формі створення теми дипломного проекту – рисунок 3.3. Після редагування Потрібно натиснути на кнопку «Зберегти».

© Мій диплом 2019

Створено студентом групи ІТ-51, Гоґалтером Русланом

Рисунок 3.3 – Редагування теми дипломного проекту

Для видалення теми дипломного проекту, потрібно на сторінці теми дипломного проекту натиснути на «Видалити». В результаті видалення викладача перенаправиться на сторінку списку власних тем дипломних проєктів – рисунок 3.4.

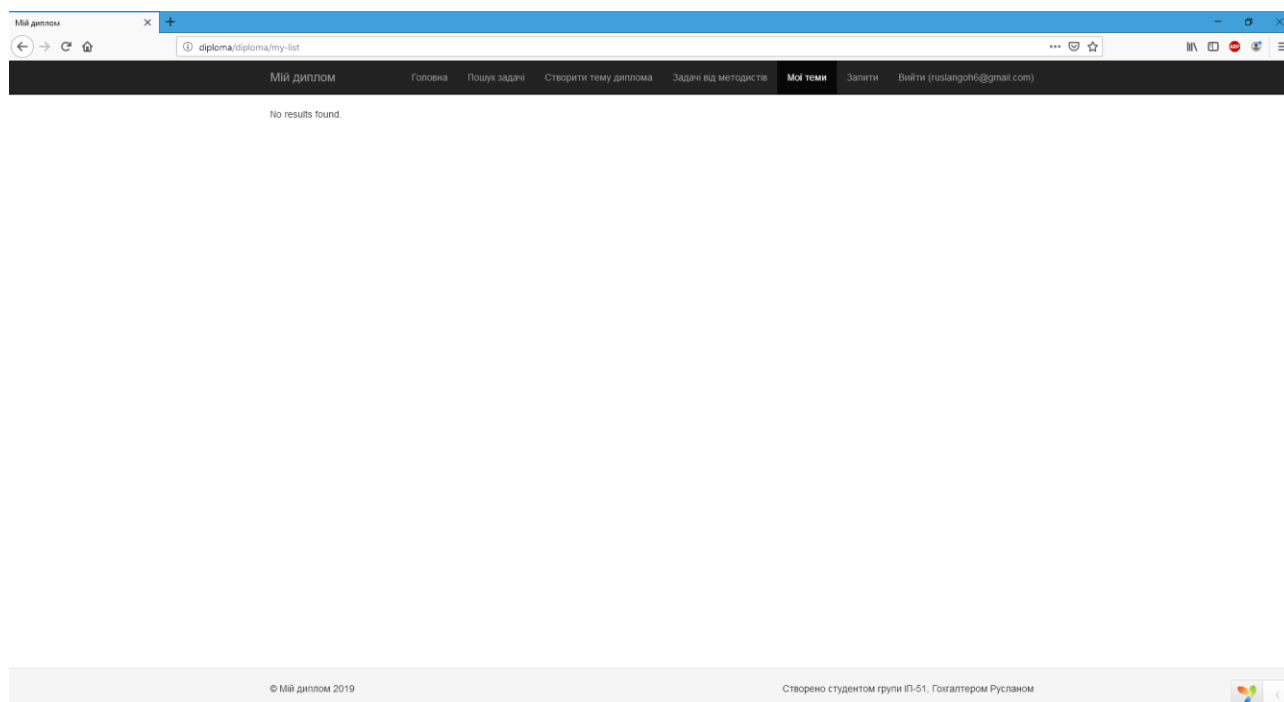


Рисунок 3.4 – Видалення теми дипломного проекту

3.2 Створення, редагування, видалення теми дипломного проекту на основі задачі від методиста

Для створення дипломного проекту на основі задачі від методиста викладач повинен натиснути на пункт меню «Задачі від методистів» - рисунок 3.5. Далі відкрити обрану задачу, натиснувши на «Відкрити» – рисунок 3.6. І натиснути на кнопку «Створити тему диплома» - рисунок 3.7. Потім, викладач редагує задачу у відповідності з умовами створення теми дипломного проекту і натискає на «Зберегти». В результаті тема диплома з'явиться у списку тем дипломних проектів – рисунок 3.8.

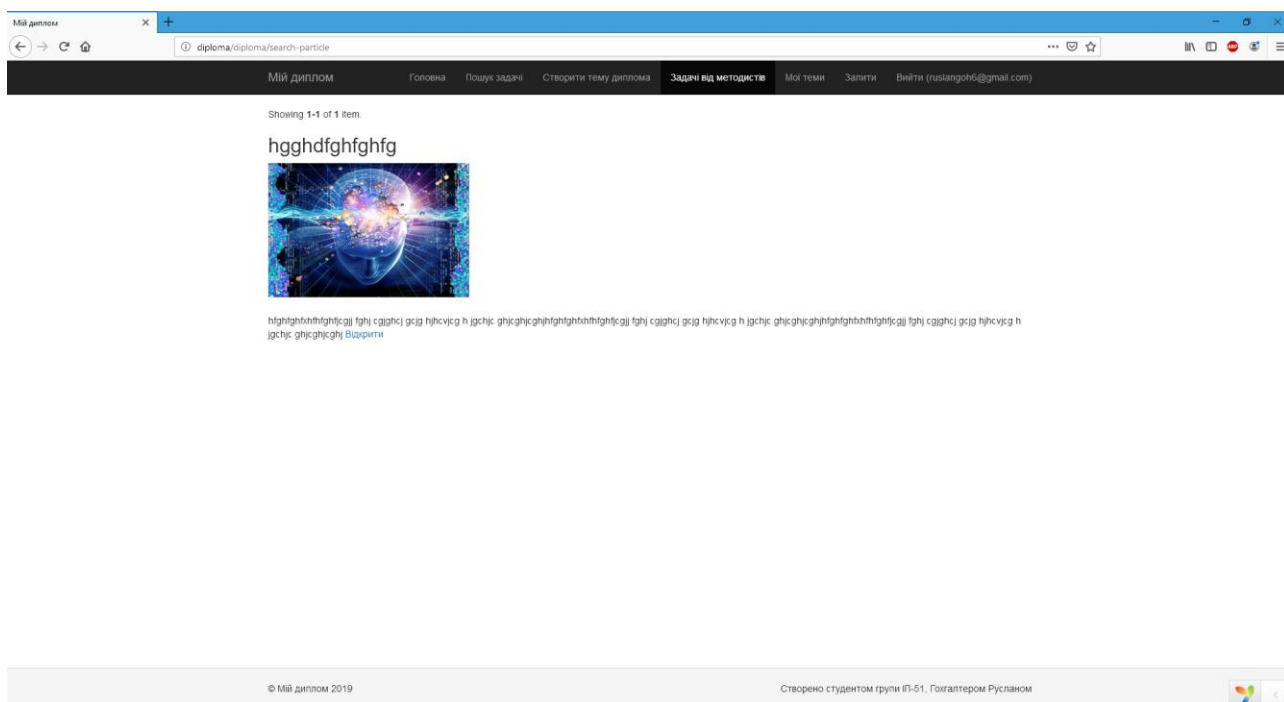


Рисунок 3.5 – Список задач від методистів

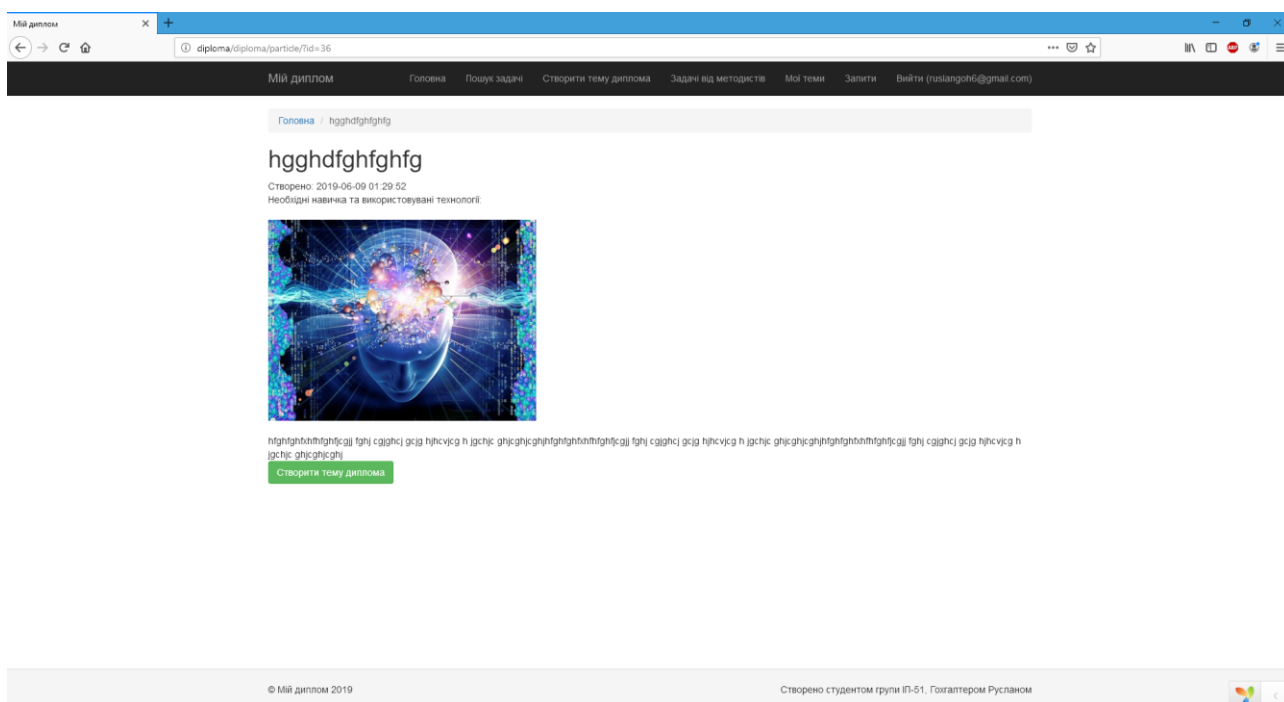


Рисунок 3.6 – Обрана задача методиста

© Мій диплом 2019

Створено студентом групи ІТ-51, Гогаггером Русланом



 

Рисунок 3.7 – Створення теми диплома на основі задачі

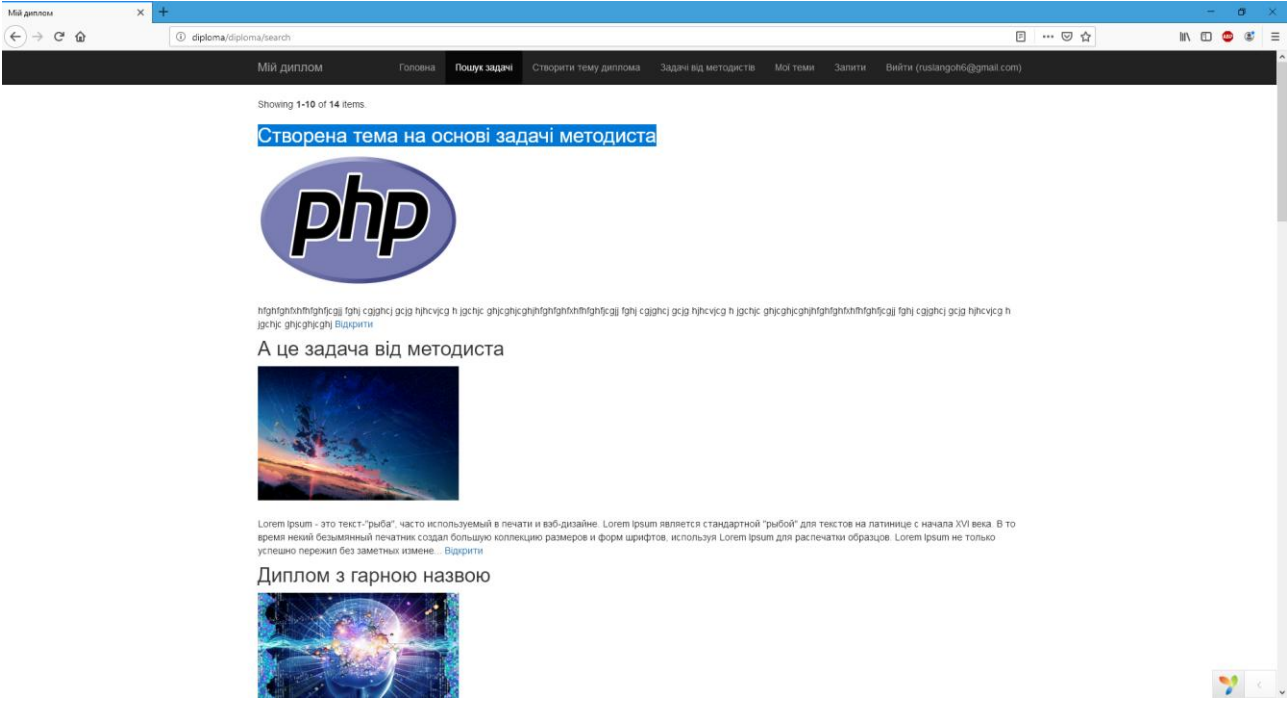


Рисунок 3.8 – Тема диплома створена на основі задачі від методиста

Видалення і редагування тем дипломів створених на основі задач від методистів, аналогічне видаленню і редагуванню власноруч створеної теми диплому.

3.3 Перегляд власних тем дипломних проектів

Для перегляду власних тем дипломних преєктів, викладачу достатньо натиснути на пункт меню «Мої теми» - рисунок 3.9.

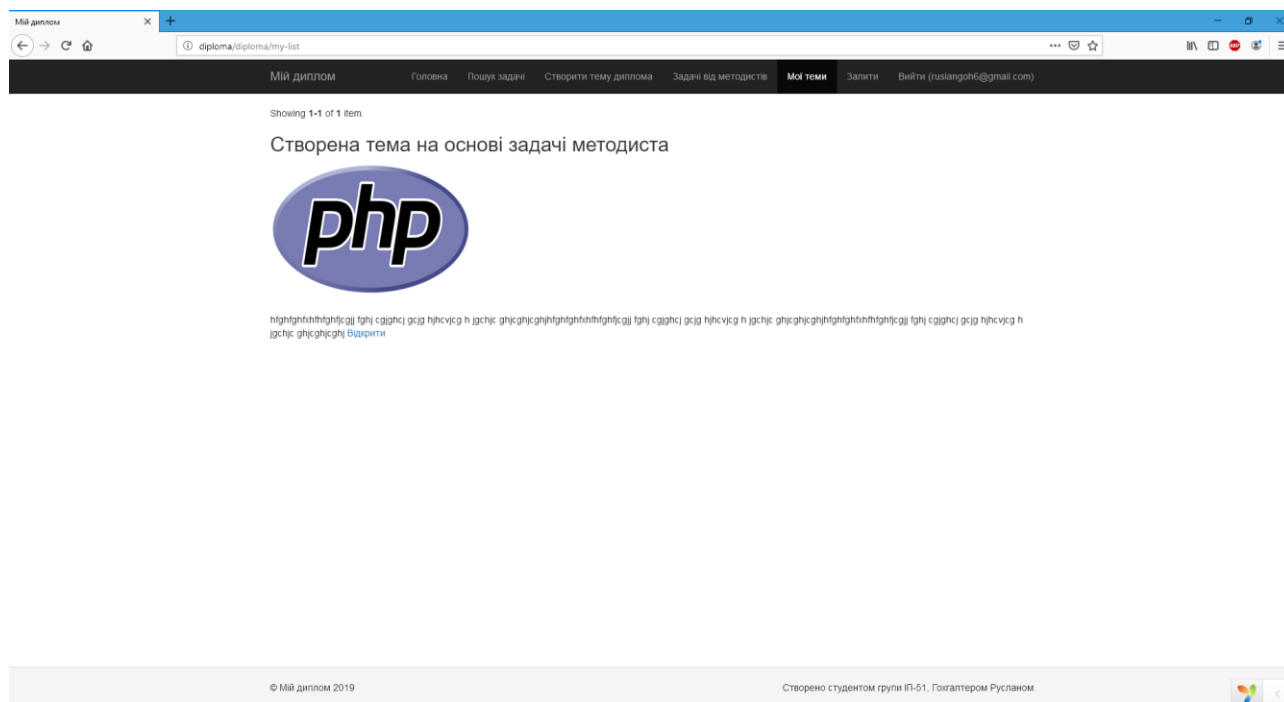


Рисунок 3.9 – Список власних тем дипломних проектів

3.4 Управління запитами на утвердження теми дипломного проекту

Для управління запитами на утвердження теми дипломного проекту, викладач повинен натиснути на кнопку «Запити». Відкриється сторінка зі всіма запитами на утвердження тем дипломних проектів викладача – рисунок 3.10.

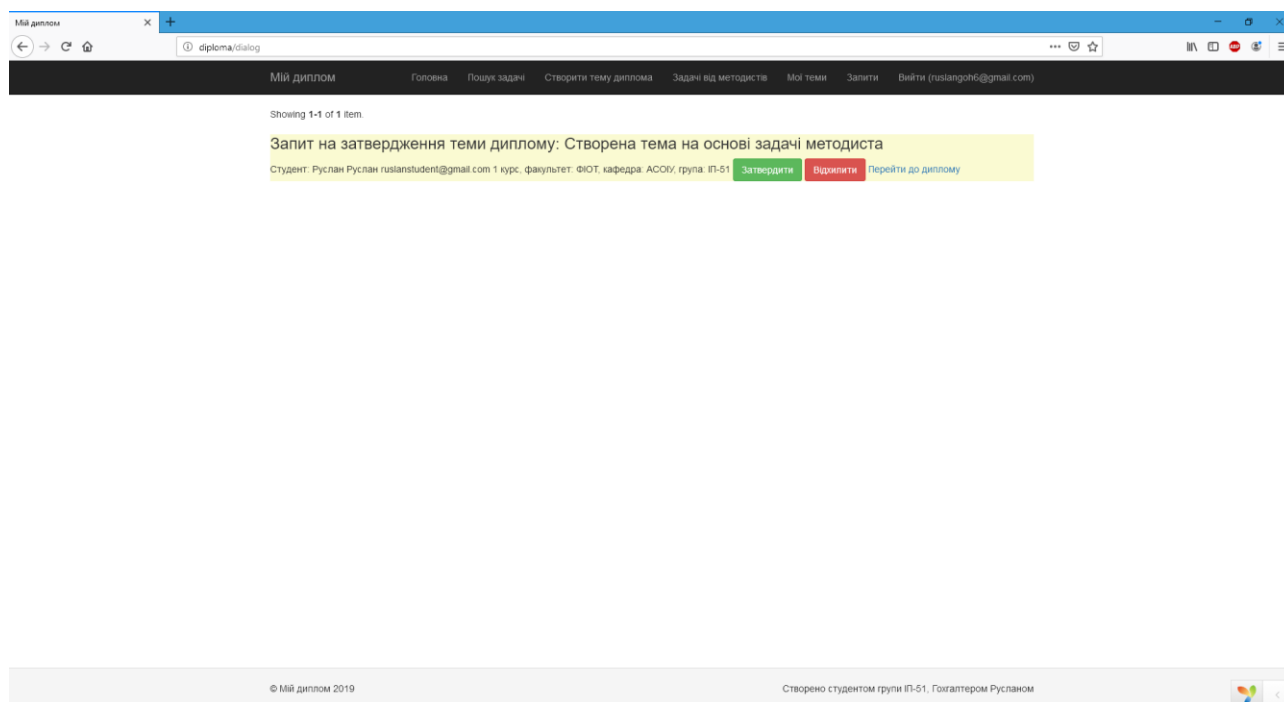


Рисунок 3.10 – Запити на утвердження тем дипломів викладача

3.4.1 Відхилення запиту на утвердження дипломної тематики

Для відхилення запиту викладачу достатньо натиснути на кнопку «Відхилити». В результаті запит позначається, як відхилений – рисунок 3.11.

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		20

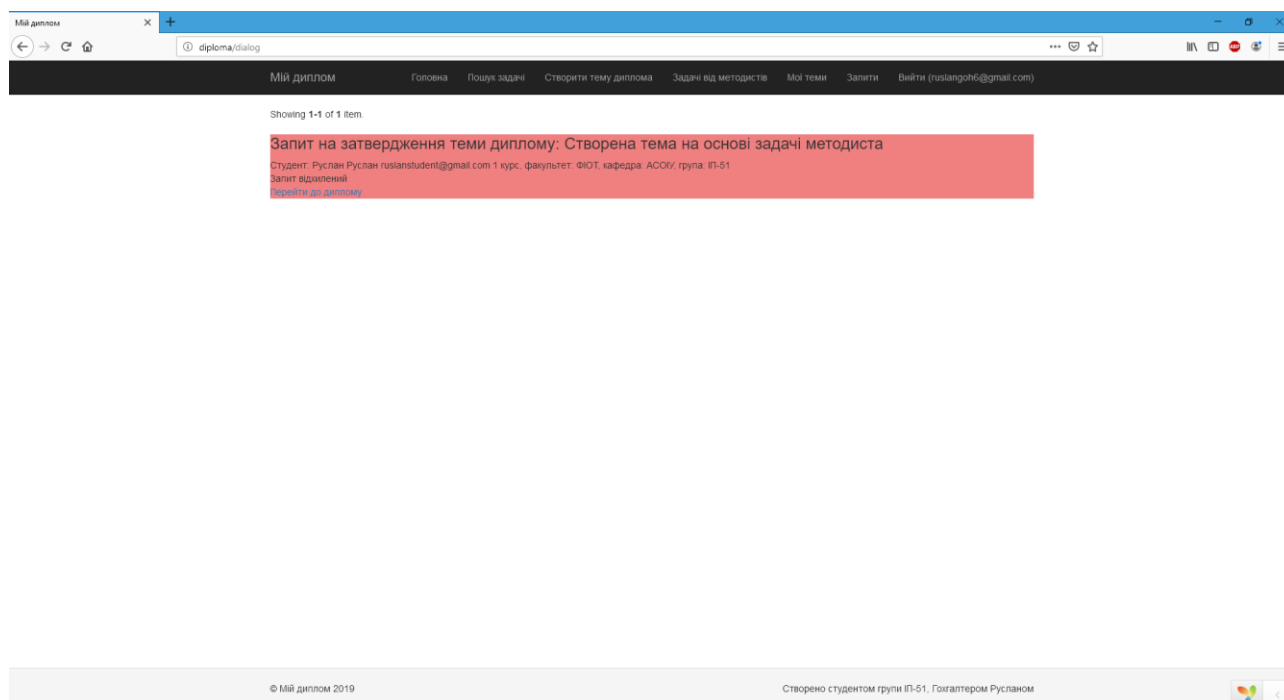


Рисунок 3.11 – Відхилений запит на утвердження теми дипломного проекту

3.4.2 Утвердження теми дипломного проекту

Для утвердження теми, викладачу достатньо натиснути на кнопку «Прийняти». В результаті запит позначається, як прийнятий – рисунок 3.12.

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		21

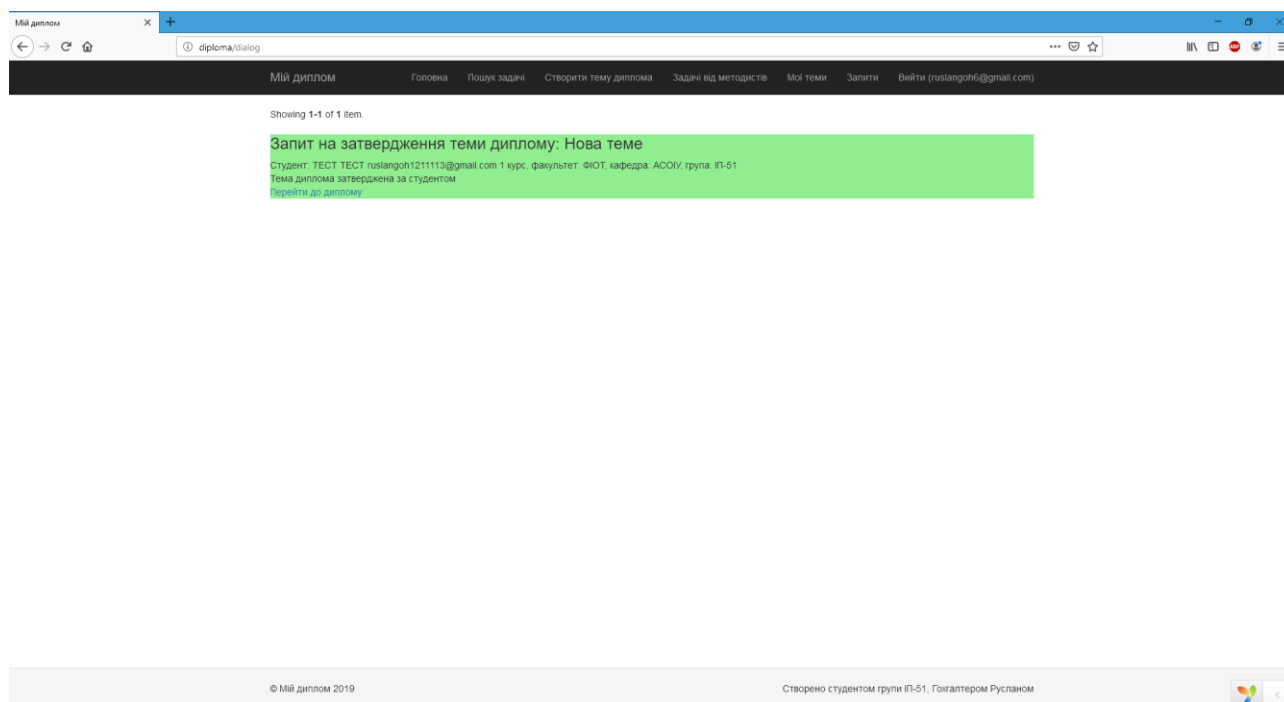


Рисунок 3.12 – Утвердження теми дипломного проекту

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

4 КЕРІВНИЦТВО МЕТОДИСТА

Методист має всі функції не зареєстрованого користувача та такі можливості на сайті:

- створення, редагування, видалення задачі;
- перегляд статусів задач і запитів.

4.1 Створення задачі

Для створення задачі методисту достатньо натиснути на пункт меню «Створити задачу». В результаті відкриється форма створення задачі – рисунок 4.1.

Рисунок 4.1 – Створення задачі

Після заповнення форми потрібно натиснути на кнопку «Створити». В результаті відкриється створена задача – рисунок 4.2.

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		23

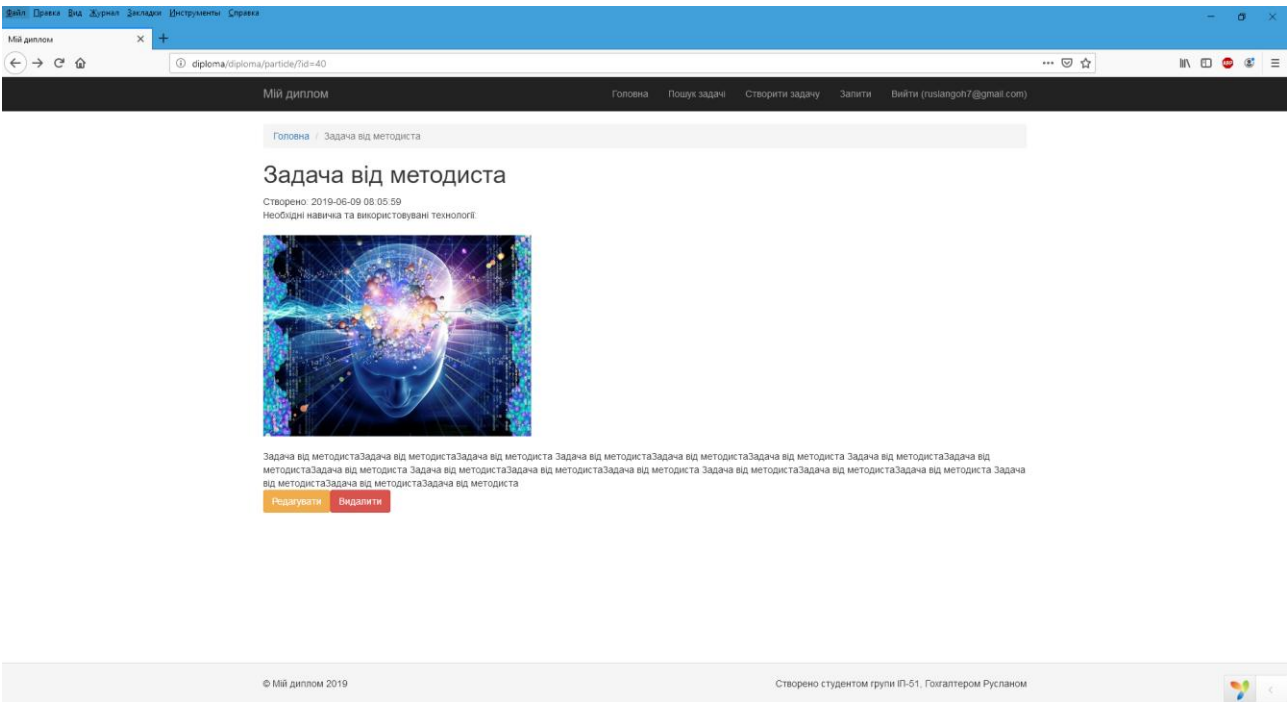


Рисунок 4.2 – Створена задача

4.2 Редагування задачі

Для редагування задачі методисту достатньо натиснути на кнопку «Редагувати» у обраній задачі. В результаті відкриється форма редагування задачі – рисунок 4.3.

Мій диплом

Головна Пошук задачі Створити задачу Заліти Вийти (ruslangoh7@gmail.com)

Головна / Створення задачі

Створення задачі

Введіть дані задачі:

Назва

Опис (від 200 до 2000 символів)

Створити

© Мій диплом 2019

Створено студентом групи ІП-51, Гогаптером Русланом

Рисунок 4.3 – Редагування задачі

Після заповнення форми потрібно натиснути на кнопку «Зберегти». В результаті відкриється створена задача – рисунок 4.4.

Мій диплом

Головна Пошук задачі Створити задачу Заліти Вийти (ruslangoh7@gmail.com)

Головна / Задача від методиста

Задача від методиста

Створено: 2019-06-09 08:05:59

Необхідні навички та використовувані технології

Задача від методиста

Редагувати Видалити

© Мій диплом 2019

Створено студентом групи ІП-51, Гогаптером Русланом

Рисунок 4.4 – Створена задача

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		25

4.3 Видалення задачі

Для видалення задачі методисту достатньо натиснути на кнопку «Видалити» у обраній задачі. В результаті задача видаляється та методист перенаправляється на список задач – рисунок 3.28.

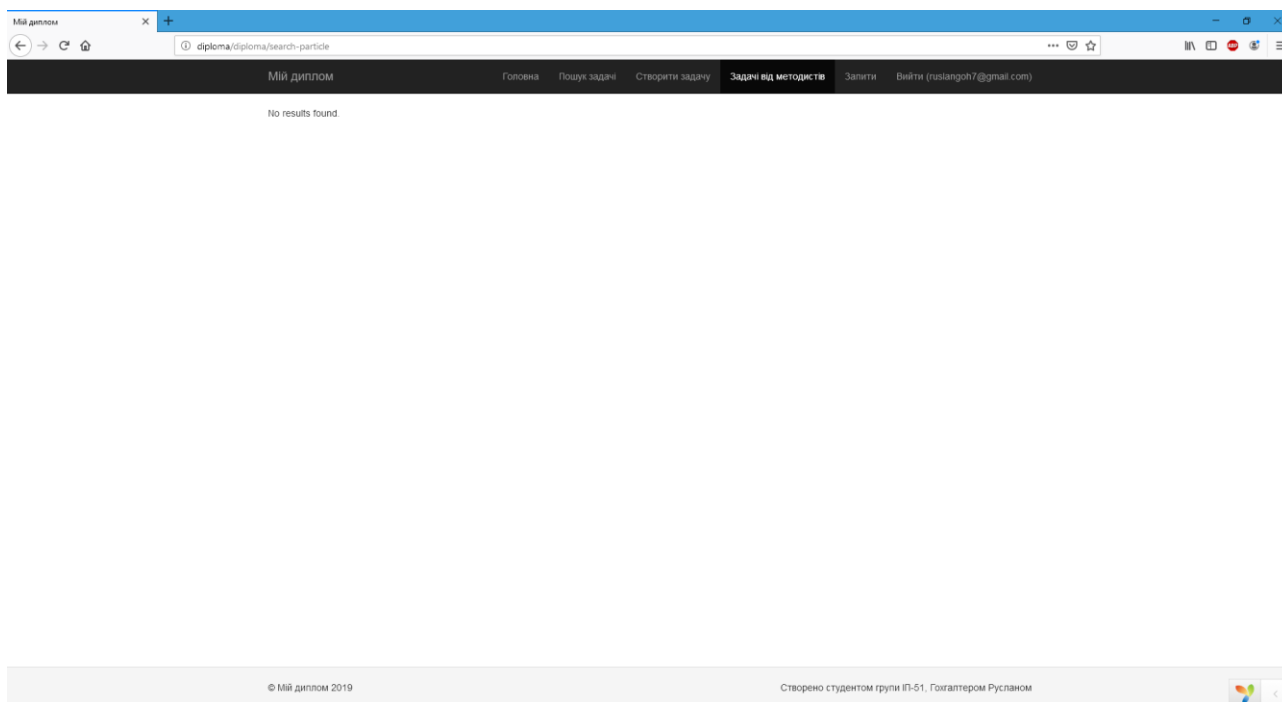


Рисунок 4.4 – Список задач

4.4 Перегляд статусу запитів

Щоб переглянути статуси задач, методист повинен натиснути на пункт меню «Запити». Відкриється список статусів запитів – рисунок 4.5.

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

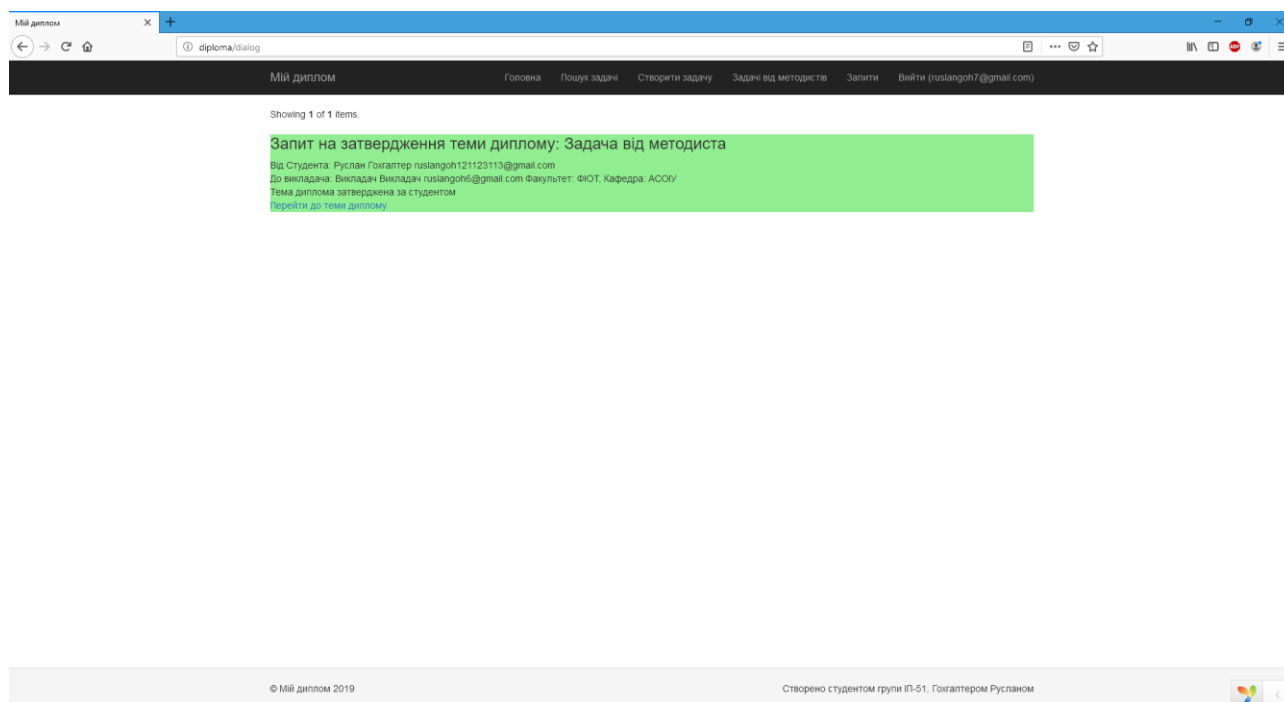


Рисунок 4.5 – Статуси запитів

					КПІ.ІП-5105.045440.06.34	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		27

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Web – застосування для узгодження тематики дипломних проектів

Керівництво адміністратора

КПІ.ІІ-5105.045440.06.33

“ПОГОДЖЕНО”

Керівник проекту:

_____ О.Д. Фіногенов

Нормоконтроль:

_____ М.М. Головченко

Виконавець:

_____ Р.С. Гохгалтер

Київ – 2019 року

ЗМІСТ

1	КЕРІВНИЦТВО АДМІНІСТРАТОРА	3
1.1	Вхід у обліковий запис	3
1.2	Управління користувачами.....	4
1.2.1	Блокування користувача	5
1.2.2	Розблокування користувача.....	6
1.2.3	Видалення користувача.....	7
1.2.4	Підтвердження реєстрації	8

1 КЕРІВНИЦТВО АДМІНІСТРАТОРА

Роль адміністратора призначена для управління користувачами. Адміністратор може передивлятися список зареєстрованих користувачів сайту та виконувати такі операції з ними:

- підтвердити реєстрацію користувача;
- заблокувати користувача;
- разблокувати користувача;
- видалити користувача.

Для доступу до цих функцій адміністратор повинен зайти до свого облікового запису, який був створений раніше.

1.1 Вхід у обліковий запис

Спочатку потрібно відкрити будь-який браузер та перейти за посиланням у сайт, рисунок 1.1.

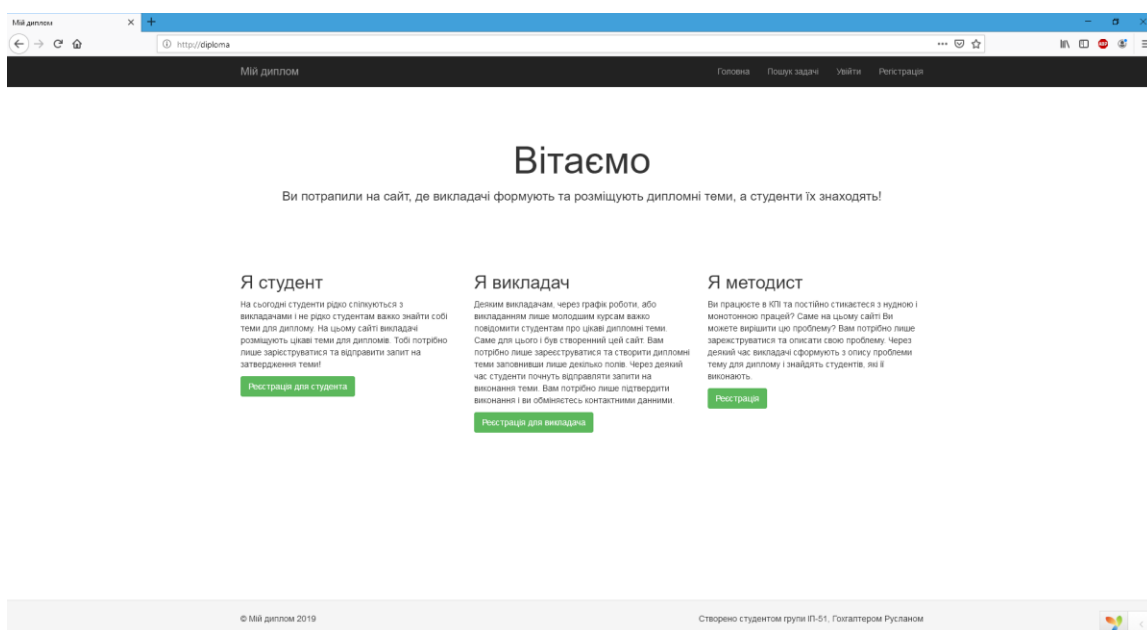


Рисунок 1.1 – Сайт

					КПІ.ІП-5105.045440.06.33	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		3

Далі потрібно натиснути на пункт меню «Увійти». Відкриється сторінка авторизації користувача – рисунок 1.2.

Рисунок 1.2 – Сторінка авторизації

Користувач повинен ввести електронну адресу і пароль адміністратора і натиснути на кнопку «Авторизація».

1.2 Управління користувачами

Після авторизації адміністратор отримує доступ до управління користувачами сайту – рисунок 1.3.

					КПІ.ІП-5105.045440.06.33	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		4

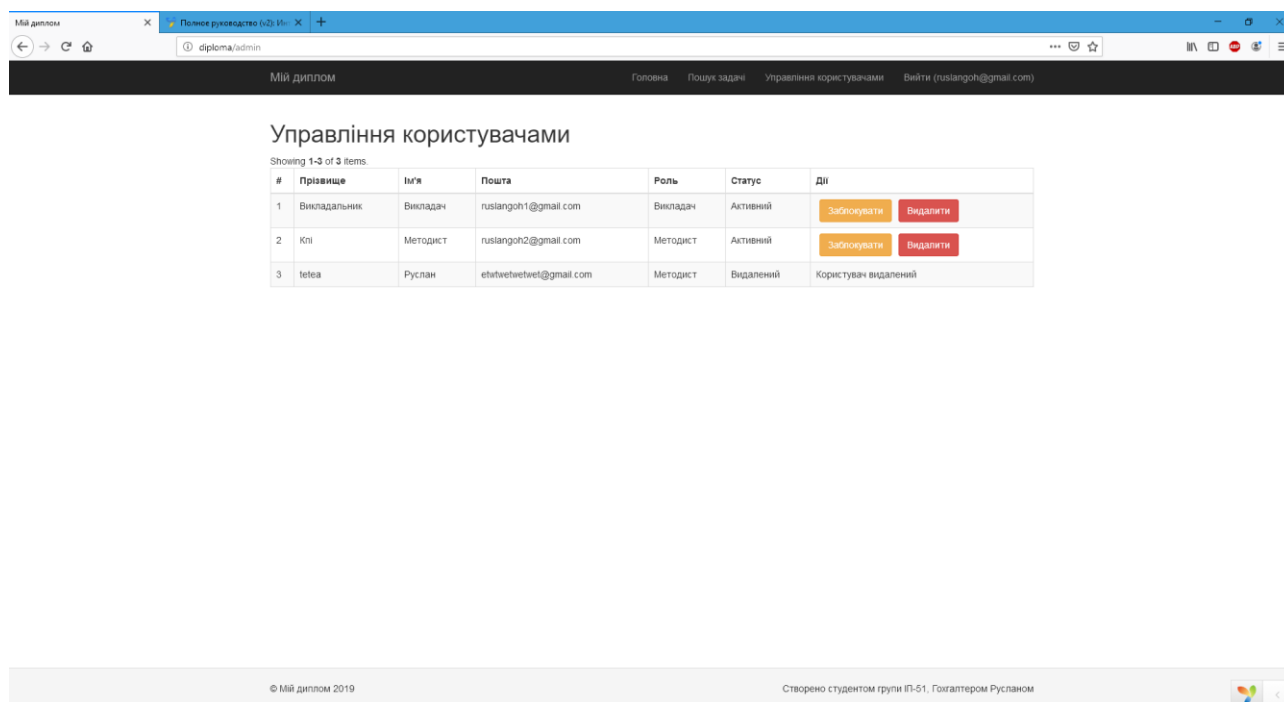


Рисунок 1.3 – Управління користувачами

1.2.1 Блокування користувача

Що заблокувати користувача, потрібно обрати користувача та натиснути на кнопку «Заблокувати». В результаті, користувач буде заблокований – рисунок 1.4.

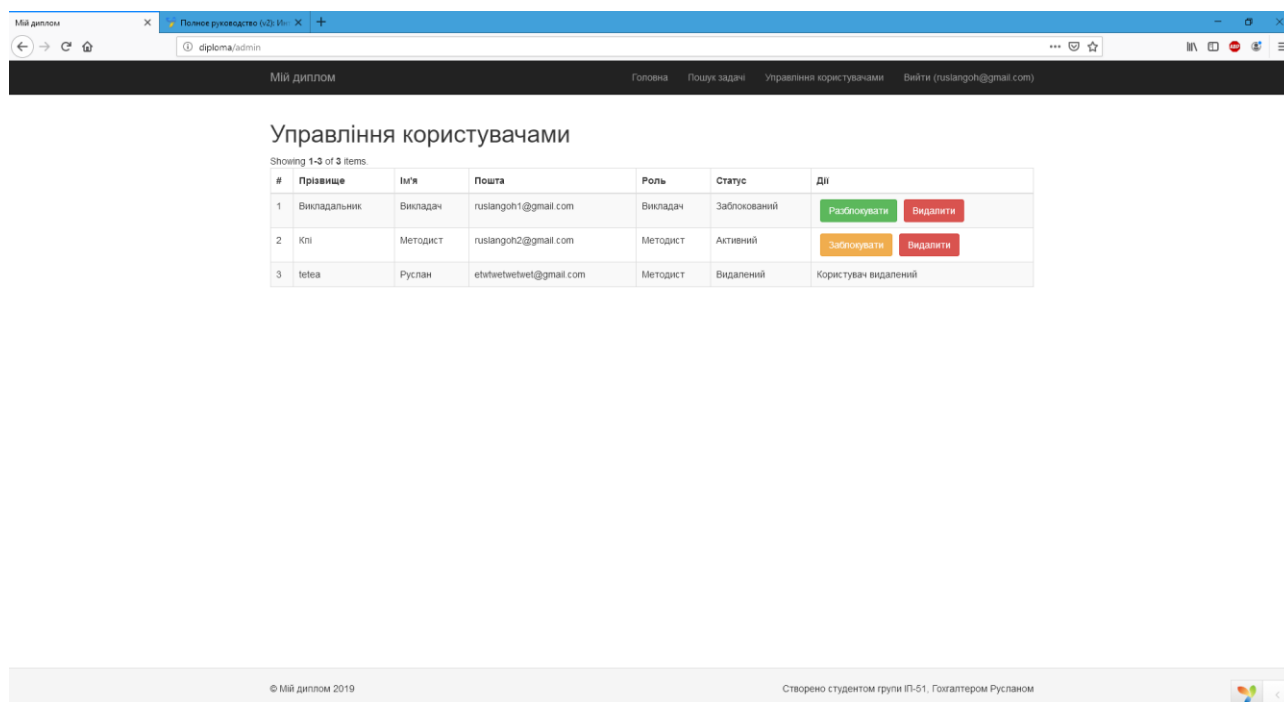


Рисунок 1.4 – Блокування користувача

1.2.2 Розблокування користувача

Що розблокувати користувача, потрібно обрати заблокованого користувача та натиснути на кнопку «Розблокувати». В результаті, користувач буде розблокований – рисунок 1.5.

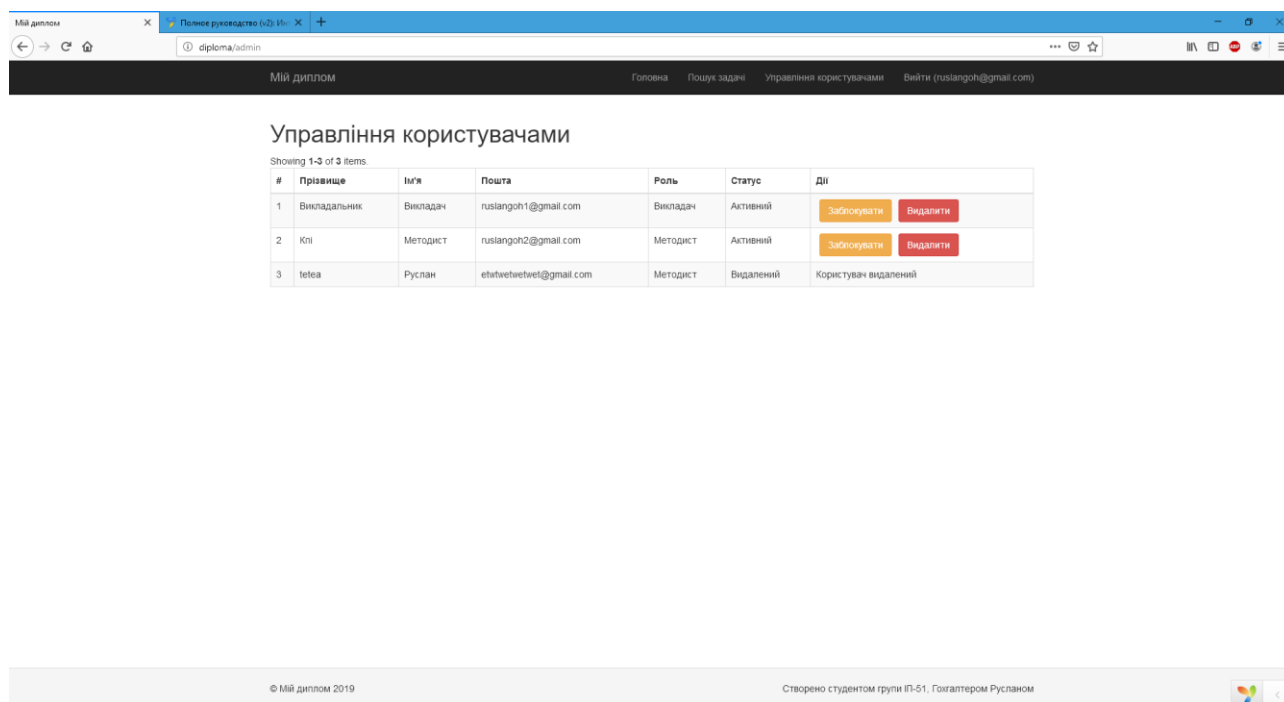


Рисунок 1.5 – Разблокування користувача

1.2.3 Видалення користувача

Що видалити користувача, потрібно обрати користувача та натиснути на кнопку «Видалити». В результаті, користувач буде видалений – рисунок 1.6.

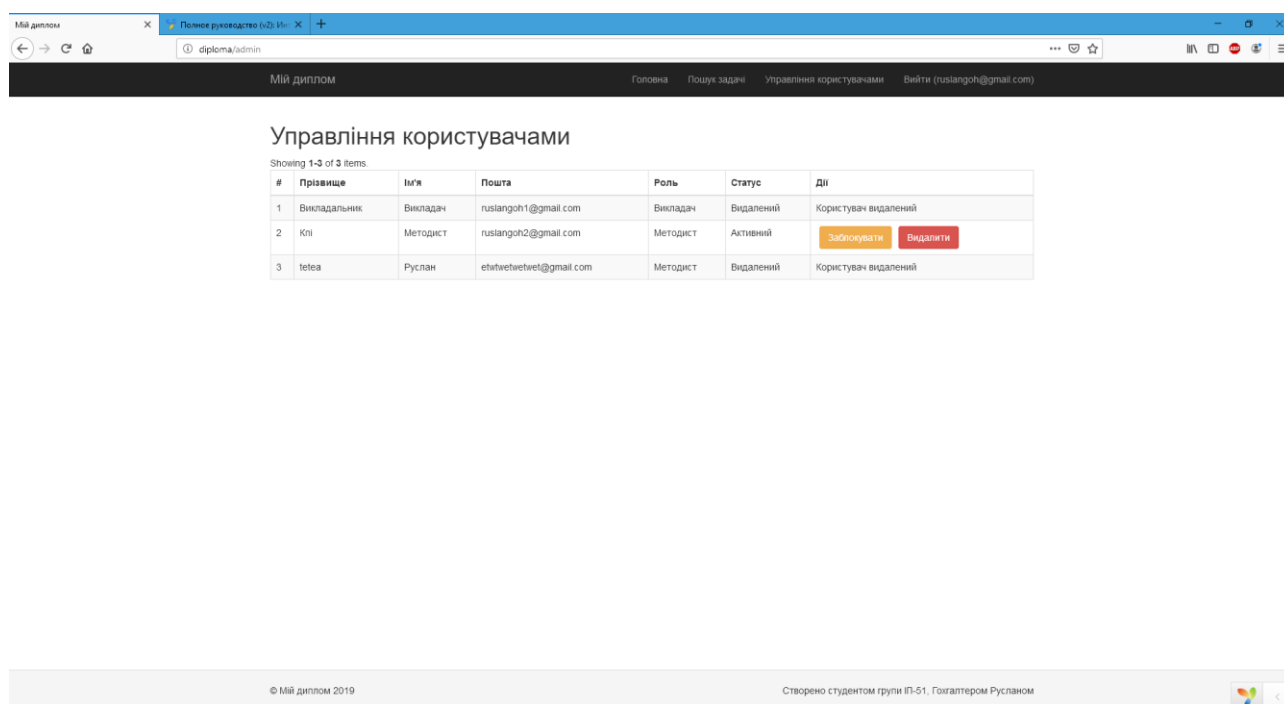


Рисунок 1.6 – Видалення користувача

1.2.4 Підтвердження реєстрації

Щоб підтвердити реєстрацію користувача, потрібно обрати користувача та натиснути на кнопку «Підтвердити». В результаті, користувач буде підтверджений – рисунок 1.7.

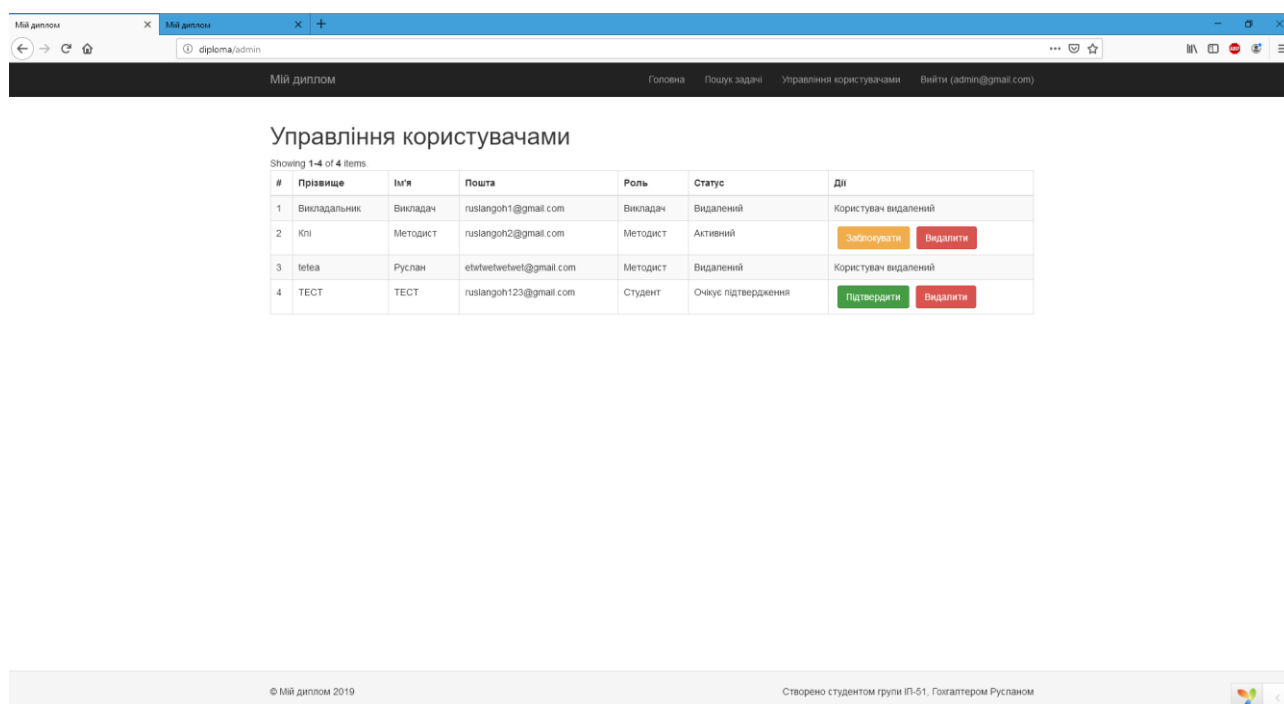


Рисунок 1.7 – Підтвердження користувача

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Web – застосування для узгодження тематики дипломних проектів

Опис програми

КПІ.ІІ-5105.045440.03.13

“ПОГОДЖЕНО”

Керівник проекту:

_____ О.Д. Фіногенов

Нормоконтроль:

_____ М.М. Головченко

Виконавець:

_____ Р.С. Гохгалтер

Київ – 2019 року

Тексти програмного коду
Web – застосування для узгодження тематики

дипломних проектів

(Найменування програми (документа))

DVD-R

(Вид носія даних)

12 арк, 212221 Кб

(Обсяг програми (документа) , арк.,) Кб)

					КПІ.ІП-5105.045440.03.13	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		2


```

<?php

namespace app\models;

use phpDocumentor\Reflection\File;
use yii\db\ActiveRecord;
use yii\data\ActiveDataProvider;
use Yii;

/**
 * Class Diploma
 *
 * @package app\models
 * @property integer $id
 * @property string $title
 * @property string $description
 * @property string $skills
 * @property integer $user_id
 * @property integer $status
 * @property integer $type
 * @property integer $worker_id
 * @property File $img
 * @property string $created_at
 * @property string $updated_at
 */
class Diploma extends ActiveRecord
{
    const STATUS_ACTIVE = 1;
    const STATUS_APPROVED = 2;
    const STATUS_DELETED = 3;
    const STATUS_PART = 4;

```

```

public $formName;

public function formName()
{
    return $this->formName ?: parent::formName();
}

public static function tableName()
{
    return 'diploma';
}

public static function createList()
{
    return new ActiveDataProvider([
        'query' => Diploma::find()->where(['status'=>1])->orderBy('created_at
DESC'),
        'pagination' => [
            'pageSize' => 10,
        ],
    ]);
}

public static function createParticleList()
{
    return new ActiveDataProvider([
        'query' => Diploma::find()->where(['status' => 4])->orderBy('created_at
DESC'),
        'pagination' => [

```

```

        'pageSize' => 10,
    ],
    ];
}

public static function changeStatus($id, $status) {
    Yii::$app->db->createCommand(
        'UPDATE ' . self::tableName() . '
        SET
            status = ' . $status . '
        WHERE id = ' . $id
    )->execute();
    return true;
}

public static function getDiplomasByTeacher() {
    if (Yii::$app->user->identity->ifTeacher()) {
        return new ActiveDataProvider([
            'query' => Diploma::find()->where(['status'=>1])-
>andWhere(['user_id' => Yii::$app->user->identity->id])->orderBy('created_at
DESC'),
            'pagination' => [
                'pageSize' => 10,
            ],
        ]);
    } else {
        exit();
    }
}
}

```

}

<?php

/**

* Created by PhpStorm.

* User: Ruslan

* Date: 24.05.2019

* Time: 22:57

*/

namespace app\models;

use yii\data\SqlDataProvider;

use yii\db\ActiveQuery;

use yii\db\ActiveRecord;

use yii\db\Query;

use Yii;

use yii\data\ActiveDataProvider;

class Dialog extends ActiveRecord

{

const STATUS_REQUEST = 1;

const STATUS_APPROVAL = 2;

const STATUS_DECLINED = 3;

const STATUS_CREATED = 4;

public static function tableName()

{

```

        return 'dialog';
    }

    public static function createTeacherDialog()
    {
        $user = Yii::$app->user->identity;
        $query = "
            SELECT
                R.`id`,
                U.`name`,
                U.`surname`,
                U.`email`,
                U.`phone`,
                G.`name` as 'gname',
                DE.`name` as 'dname',
                F.`name` as 'fname',
                S.`curse`,
                U.`gender`,
                D.`title`,
                D.`status` as 'diploma_status',
                R.`status` as 'dialog_status',
                D.`id` as 'diploma_id'
            FROM `dialog` AS R
            LEFT JOIN `diploma` AS D ON R.`diploma_id` = D.`id`
            LEFT JOIN `user` AS U ON U.`id` = R.`user_id`
            LEFT JOIN `student` AS S ON S.`user_id` = U.`id`
            LEFT JOIN `group` AS G ON G.`id` = S.`group_id`
            LEFT JOIN `departament` AS DE ON DE.`id` = G.`departament_id`
            LEFT JOIN `faculty` AS F ON F.`id` = DE.`faculty_id`
        ";
    }

```

```
WHERE D.`user_id` = " . $user->id . "
AND R.`status` != 4
ORDER BY R.`created_at`, R.`updated_at`
";
```

```
$dataProvider = new SqlDataProvider([
    'sql' => $query,
    'pagination' => [
        'pageSize' => 10,
    ],
]);
return $dataProvider;
}
```

```
public static function createStudentDialog()
{
    $user = Yii::$app->user->identity;
    $query = "
        SELECT
            R.`id`,
            U.`name`,
            U.`surname`,
            U.`email`,
            U.`phone`,
            DE.`name` as 'dname',
            F.`name` as 'fname',
            T.`degree`,
            U.`gender`,
            D.`title`,
```

```

        D.`status` as 'diploma_status',
        R.`status` as 'dialog_status',
        D.`id` as 'diploma_id'
FROM `dialog` AS R
LEFT JOIN `diploma` AS D ON R.`diploma_id` = D.`id`
LEFT JOIN `user` AS U ON U.`id` = D.`user_id`
LEFT JOIN `teacher` AS T ON T.`user_id` = U.`id`
LEFT JOIN `departament` AS DE ON DE.`id` = T.`departament_id`
LEFT JOIN `faculty` AS F ON F.`id` = DE.`faculty_id`
WHERE R.`user_id` = " . $user->id . "
ORDER BY R.`created_at`, R.`updated_at`
";

```

```

$dataProvider = new SqlDataProvider([
    'sql' => $query,
    'pagination' => [
        'pageSize' => 10,
    ],
]);
return $dataProvider;
}

```

```

public static function createWorkerDialog()
{
    $user = Yii::$app->user->identity;
    $query = "
        SELECT
            R.`id`,
            U.`name`,

```

```

        U.`surname`,
        U.`email`,
        U.`phone`,
        U.`gender`,
        DE.`name` as 'dname',
        F.`name` as 'fname',
        US.`name` as 'sname',
        US.`surname` as 'ssurname',
        US.`email` as 'semail',
        US.`phone` as 'sphone',
        D.`title`,
        D.`status` as 'diploma_status',
        R.`status` as 'dialog_status',
        D.`id` as 'diploma_id'
FROM `dialog` AS R
LEFT JOIN `diploma` AS D ON R.`diploma_id` = D.`id`
LEFT JOIN `user` AS U ON U.`id` = D.`user_id`
LEFT JOIN `user` AS US ON US.`id` = R.`user_id`
LEFT JOIN `teacher` AS T ON T.`user_id` = U.`id`
LEFT JOIN `departament` AS DE ON DE.`id` = T.`departament_id`
LEFT JOIN `faculty` AS F ON F.`id` = DE.`faculty_id`
WHERE R.`user_id` = " . $user->id . " OR D.worker_id = " . $user->id
. "

ORDER BY R.`created_at` DESC
";

$dataProvider = new SqlDataProvider([
    'sql' => $query,
    'pagination' => [

```



```

        'pageSize' => 10,
    ],
    ];
    return $dataProvider;
}

public static function exist($userId, $diplomaId)
{
    return Dialog::find()->where(['diploma_id' => $diplomaId, 'user_id' =>
$userId])->exists();
}

public static function needHide($diplomaId, User $user)
{
    if (!self::exist($user->id, $diplomaId) && $user->ifStudent()) return false;
    else return true;
}

public static function changeStatus($id, $status)
{
    if ($status == 2 || $status == 3) {
        Yii::$app->db->createCommand(
            'UPDATE ' . self::tableName() . '
            SET
                status = ' . $status . '
            WHERE id = ' . $id
        )->execute();
    }
}

```

```
        return true;
    } else {
        return false;
    }
}

public static function declineAll($diploma_id)
{
    Yii::$app->db->createCommand(
        'UPDATE ' . self::tableName() . '
        SET
            status = ' . self::STATUS_DECLINED . '
        WHERE diploma_id = ' . $diploma_id . '
        AND status != 2
    ')->execute();
    return true;
}
}
```

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Web – застосування для узгодження тематики дипломних проектів

Графічний матеріал

КПІ.ІІ-5105.045440.07.99

“ПОГОДЖЕНО”

Керівник проекту:

_____ О.Д. Фіногенов

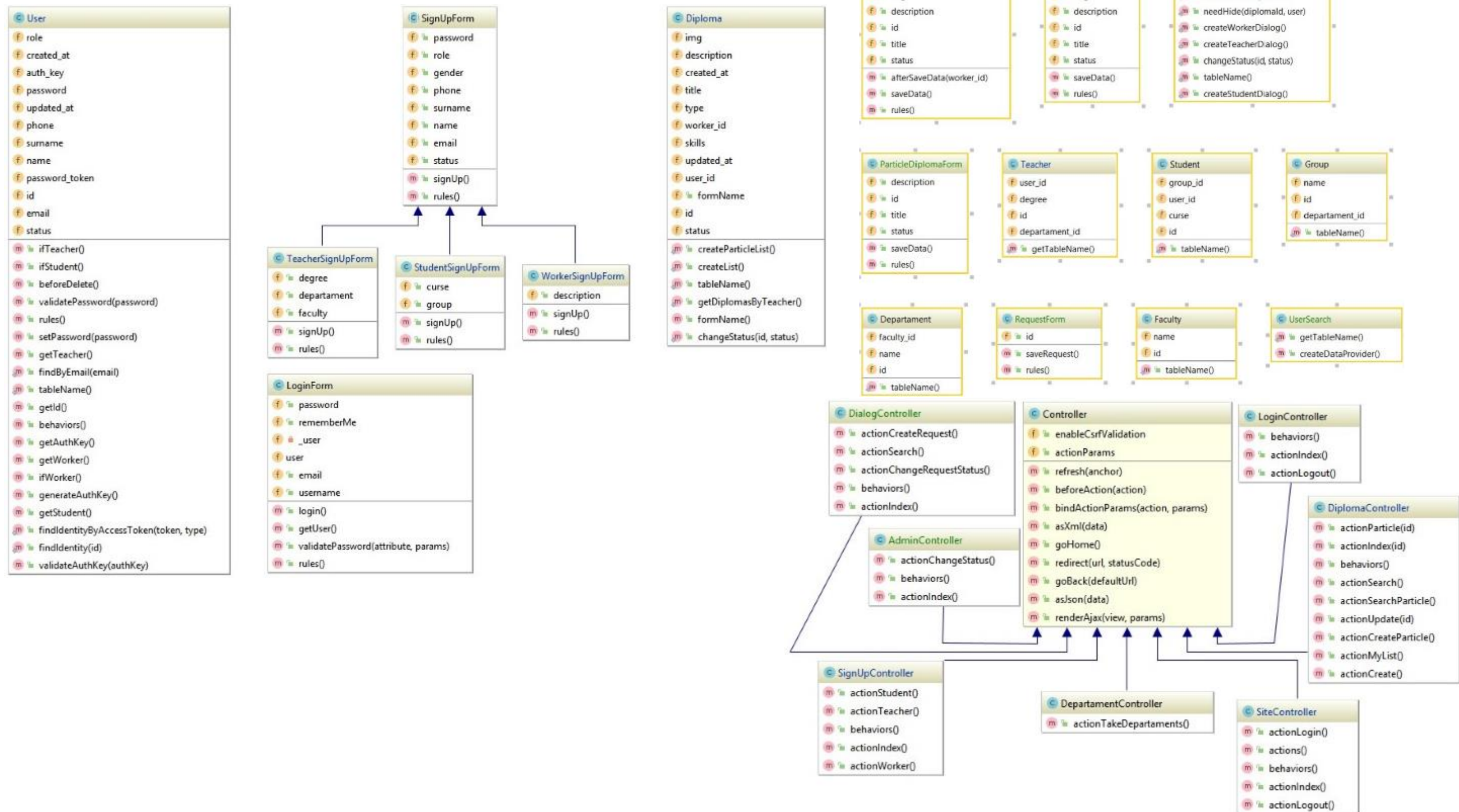
Нормоконтроль:

_____ М.М. Головченко

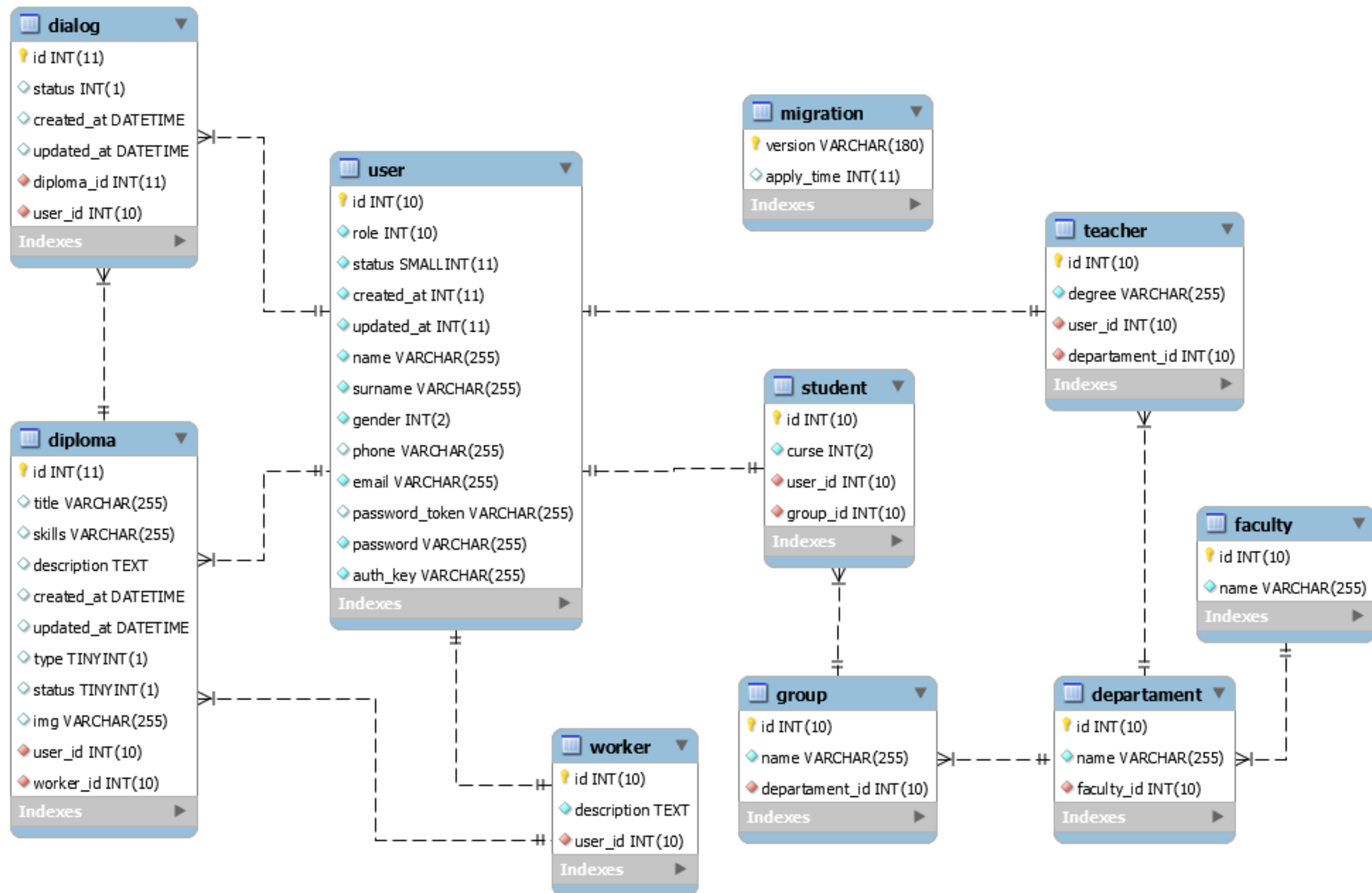
Виконавець:

_____ Р.С. Гохгалтер

Київ – 2019 року



					КПІ.ІП-5105.045440.07.99.СС			
					Схема Структурна класів програмного забезпечення	Літера	Маса	Масштаб
Зм.	Арк.	№ документа	Підпис	Дата				
Розробив		Гохгалтер Р.С						
Перевірів		Фіногенов О.Д.						
Т. кон.					Web – застосування для узгодження тематики дипломних проектів	Аркуш		Аркушів
Н. кон.		Головченко М.М.				КПІ ім.Ігоря Сікорського Кафедра АСОІУ гр. ІП-51		
Затвердив		Фіногенов О.Д.						



					КПІ.ІП-5105.045440.07.99.СБД					
					Схема бази даних	Літера			Маса	Масштаб
Зм.	Арк.	№ документа	Підпис	Дата						
Розробив	Гохгалтер Р.С									
Перевірив	Фіногенов О.Д.									
Т. кон.						Аркуш			Аркушів	
					Web – застосування для узгодження тематики дипломних проєктів	КПІ ім.Ігоря Сікорського Кафедра АСОІУ гр. ІП-51				
Н. кон.	Головченко М.М.									
Затвердив	Фіногенов О.Д.									